Pattern

Prontuario AWS

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Prontuario AWS: Pattern

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

I marchi e l’immagine commerciale di Amazon non possono essere utilizzati in relazione a prodotti o servizi che non siano di Amazon, in una qualsiasi modalità che possa causare confusione tra i clienti o in una qualsiasi modalità che denigri o discrediti Amazon. Tutti gli altri marchi non di proprietà di Amazon sono di proprietà dei rispettivi proprietari, che possono o meno essere affiliati, collegati o sponsorizzati da Amazon.

Table of Contents

AWS Modelli di guida prescrittivi 1
Fondamenti del cloud 3
Automatizza la creazione di account utilizzando Landing Zone Accelerator 4
Riepilogo 4
Prerequisiti e limitazioni 4
Architecture 5
Tools (Strumenti) 6
Best practice 9
Epiche 9
Risorse correlate 35
Informazioni aggiuntive 35
Inventaria automaticamente AWS le risorse 37
Riepilogo 37
Prerequisiti e limitazioni 37
Architecture 39
Tools (Strumenti) 39
Best practice 40
Epiche 41
Risoluzione dei problemi 48
Risorse correlate 48
Costruisci una AWS landing zone con MongoDB Atlas 50
Riepilogo 50
Prerequisiti e limitazioni 51
Architecture 51
Tools (Strumenti) 53
Epiche 53
Risorse correlate 63
Configurazione dei log di flusso VPC tra gli account 65
Riepilogo 65
Prerequisiti e limitazioni 65
Architecture 66
Tools (Strumenti) 66
Best practice 67
Epiche 69

Risorse correlate 70
Informazioni aggiuntive 70
Gestisci i set di AWS autorizzazioni utilizzando Terraform 73
Riepilogo 73
Prerequisiti e limitazioni 74
Architecture 74
Tools (Strumenti) 76
Best practice 78
Epiche 78
Risoluzione dei problemi 89
Risorse correlate 90
Informazioni aggiuntive 90
Etichetta automaticamente gli allegati Transit Gateway 92
Riepilogo 92
Prerequisiti e limitazioni 92
Architecture 93
Tools (Strumenti) 94
Epiche 95
Risorse correlate 102
Altri modelli 103
AI e apprendimento automatico 104
Associa un AWS CodeCommit repository ad Amazon SageMaker AI Studio Classic tra più account 106
Riepilogo 106
Prerequisiti e limitazioni 106
Architecture 106
Tools (Strumenti) 107
Epiche 108
Informazioni aggiuntive 113
Estrai automaticamente il contenuto dai file PDF 116
Riepilogo 116
Prerequisiti e limitazioni 116
Architecture 117
Tools (Strumenti) 118
Epiche 119
Risorse correlate 123

Allegati 124
Crea un modello di previsione con avvio a freddo utilizzando SageMaker AI DeepAr 125
Riepilogo 125
Prerequisiti e limitazioni 125
Architecture 126
Tools (Strumenti) 127
Best practice 128
Epiche 128
Risorse correlate 130
Crea un MLOps flusso di lavoro usando SageMaker AI e Azure DevOps 131
Riepilogo 131
Prerequisiti e limitazioni 131
Architecture 132
Tools (Strumenti) 134
Best practice 135
Epiche 136
risoluzione dei problemi 145
Risorse correlate 146
Configura la registrazione in Amazon Bedrock utilizzando CloudFormation 148
Riepilogo 148
Prerequisiti e limitazioni 148
Architecture 149
Tools (Strumenti) 150
Epiche 151
Risorse correlate 153
Crea contenitori Docker SageMaker per l'addestramento dei modelli in Step Functions 155
Riepilogo 155
Prerequisiti e limitazioni 155
Architecture 156
Tools (Strumenti) 156
Epiche 157
Risorse correlate 170
Crea controlli di accesso in Amazon EKS con gli agenti Amazon Bedrock 171
Riepilogo 171
Prerequisiti e limitazioni 171
Architecture 172

Tools (Strumenti) 173
Best practice 174
Epiche 174
risoluzione dei problemi 192
Risorse correlate 192
Implementa un caso d'uso RAG su AWS 194
Riepilogo 194
Prerequisiti e limitazioni 194
Architecture 195
Tools (Strumenti) 197
Best practice 198
Epiche 199
Risorse correlate 202
Informazioni aggiuntive 202
Distribuisci più oggetti del modello di pipeline in un unico endpoint SageMaker 204
Riepilogo 204
Prerequisiti e limitazioni 204
Architecture 205
Tools (Strumenti) 205
Epiche 206
Risorse correlate 216
Implementa la convalida della sicurezza della codifica in tempo reale con Kiro 217
Riepilogo 217
Prerequisiti e limitazioni 218
Architecture 219
Tools (Strumenti) 220
Best practice 222
Epiche 223
Risoluzione dei problemi 231
Risorse correlate 232
Informazioni aggiuntive 233
Sviluppa assistenti di chat basati sull'intelligenza artificiale utilizzando RAG e prompting ReAct 236
Riepilogo 236
Prerequisiti e limitazioni 237
Architecture 238

Tools (Strumenti) 239
Best practice 241
Epiche 241
Risoluzione dei problemi 249
Risorse correlate 249
Informazioni aggiuntive 250
Sviluppa un assistente basato su chat utilizzando Amazon Bedrock 251
Riepilogo 251
Prerequisiti e limitazioni 252
Architecture 252
Tools (Strumenti) 253
Best practice 255
Epiche 256
Risorse correlate 259
Informazioni aggiuntive 260
Documenta le conoscenze istituzionali grazie agli input vocali 262
Riepilogo 262
Prerequisiti e limitazioni 262
Architecture 263
Tools (Strumenti) 264
Best practice 266
Epiche 266
Risorse correlate 272
Genera consigli personalizzati con Amazon Personalize 274
Riepilogo 274
Prerequisiti e limitazioni 274
Architecture 274
Tools (Strumenti) 275
Epiche 277
Risorse correlate 279
Informazioni aggiuntive 280
Semplifica i flussi di lavoro ML utilizzando AI e Hydra SageMaker 284
Riepilogo 284
Prerequisiti e limitazioni 284
Architecture 285
Tools (Strumenti) 286

Best practice 287
Epiche 287
Risoluzione dei problemi 295
Risorse correlate 295
Informazioni aggiuntive 295
..... 297
Riepilogo 297
Prerequisiti e limitazioni 298
Architecture 301
Tools (Strumenti) 303
Best practice 304
Epiche 304
Risorse correlate 311
Informazioni aggiuntive 312
Usa Amazon Q Developer come assistente di programmazione 317
Riepilogo 317
Prerequisiti e limitazioni 317
Tools (Strumenti) 318
Best practice 318
Epiche 319
risoluzione dei problemi 327
Risorse correlate 327
Utilizzate SageMaker Processing per l'ingegneria distribuita delle funzionalità di set di dati ML su scala terabyte 328
Riepilogo 328
Prerequisiti e limitazioni 328
Architecture 329
Tools (Strumenti) 332
Epiche 332
Risorse correlate 345
Allegati 346
Visualizza i risultati AI/ML del modello utilizzando Flask ed Elastic Beanstalk 347
Riepilogo 347
Prerequisiti e limitazioni 347
Architecture 348
Tools (Strumenti) 350

Epiche 350
Risorse correlate 360
Informazioni aggiuntive 360
Altri modelli 365
Analisi 366
Analizza i dati di Amazon Redshift in Microsoft SQL Server Analysis Services 368
Riepilogo 368
Prerequisiti e limitazioni 368
Architecture 368
Tools (Strumenti) 369
Epiche 369
Risorse correlate 371
..... 372
Riepilogo 372
Prerequisiti e limitazioni 372
Architecture 373
Tools (Strumenti) 373
Epiche 374
Risorse correlate 379
Automatizza l'inserimento dei dati da Amazon S3 AWS Data Exchange 380
Riepilogo 380
Prerequisiti e limitazioni 380
Architecture 380
Tools (Strumenti) 381
Epiche 382
Risorse correlate 384
Allegati 385
Automatizza l'applicazione della crittografia in AWS Glue 386
Riepilogo 386
Prerequisiti e limitazioni 386
Architecture 386
Tools (Strumenti) 387
Best practice 388
Epiche 388
Risorse correlate 391

Crea una pipeline di dati per elaborare i dati di Google Analytics utilizzando il Development Kit AWS DataOps ... 392
Riepilogo ... 392
Prerequisiti e limitazioni ... 392
Architecture ... 393
Tools (Strumenti) ... 393
Epiche ... 394
risoluzione dei problemi ... 396
Risorse correlate ... 397
Informazioni aggiuntive ... 397
Crea una pipeline di elaborazione video ... 400
Riepilogo ... 400
Prerequisiti e limitazioni ... 400
Architecture ... 401
Tools (Strumenti) ... 401
Epiche ... 402
Risorse correlate ... 409
Informazioni aggiuntive ... 410
Allegati ... 410
Crea una pipeline ETL da Amazon S3 ad Amazon Redshift ... 411
Riepilogo ... 411
Prerequisiti e limitazioni ... 411
Architecture ... 412
Tools (Strumenti) ... 412
Epiche ... 413
Risorse correlate ... 420
Informazioni aggiuntive ... 421
Crea una rete dati aziendale con Amazon DataZone ... 422
Riepilogo ... 422
Prerequisiti e limitazioni ... 422
Architecture ... 423
Tools (Strumenti) ... 424
Epiche ... 425
Risorse correlate ... 438
Informazioni aggiuntive ... 439
Calcola il valore a rischio (VaR) utilizzando i servizi AWS ... 441

Riepilogo 441
Prerequisiti e limitazioni 441
Architecture 442
Tools (Strumenti) 443
Best practice 444
Epiche 444
Risorse correlate 448
Configura l'accesso tra account a un catalogo dati AWS Glue condiviso 449
Riepilogo 449
Prerequisiti e limitazioni 449
Architecture 449
Tools (Strumenti) 451
Epiche 451
Risorse correlate 464
Informazioni aggiuntive 464
Convertire NORMALIZE in Amazon Redshift SQL 465
Riepilogo 465
Prerequisiti e limitazioni 465
Architecture 465
Tools (Strumenti) 466
Epiche 471
Risorse correlate 471
Converti RESET WHEN in Amazon Redshift SQL 473
Riepilogo 473
Prerequisiti e limitazioni 473
Architecture 473
Tools (Strumenti) 474
Epiche 478
Risorse correlate 478
Implementa e gestisci un data lake serverless su AWS 480
Riepilogo 480
Prerequisiti e limitazioni 480
Architecture 480
Tools (Strumenti) 482
Epiche 483
Risorse correlate 485

..... 487
Riepilogo 487
Prerequisiti e limitazioni 487
Architecture 488
Tools (Strumenti) 488
Epiche 489
Risorse correlate 492
Allegati 492
Garantisci la registrazione di Amazon EMR su Amazon S3 493
Riepilogo 493
Prerequisiti e limitazioni 494
Architecture 494
Tools (Strumenti) 495
Epiche 495
Risorse correlate 498
Allegati 498
Genera dati di test con AWS Glue 499
Riepilogo 499
Prerequisiti e limitazioni 499
Architecture 500
Tools (Strumenti) 500
Best practice 500
Epiche 501
Risorse correlate 511
Informazioni aggiuntive 511
Inserisci dati IoT direttamente in Amazon S3 516
Riepilogo 516
Prerequisiti e limitazioni 516
Architecture 517
Tools (Strumenti) 518
Best practice 518
Epiche 519
risoluzione dei problemi 527
Risorse correlate 527
Informazioni aggiuntive 528
Avvia un job Spark in Amazon EMR utilizzando una funzione Lambda 532

Riepilogo 532
Prerequisiti e limitazioni 532
Architecture 533
Tools (Strumenti) 533
Epiche 534
Risorse correlate 537
Informazioni aggiuntive 537
Allegati 540
Esegui la migrazione dei carichi di lavoro Apache Cassandra su Amazon Keyspaces 541
Riepilogo 541
Prerequisiti e limitazioni 541
Architecture 541
Tools (Strumenti) 542
Best practice 543
Epiche 543
risoluzione dei problemi 557
Risorse correlate 557
Informazioni aggiuntive 558
Esegui la migrazione di Oracle Business Intelligence 12C al cloud AWS 559
Riepilogo 559
Prerequisiti e limitazioni 559
Architecture 560
Tools (Strumenti) 561
Epiche 562
Risorse correlate 574
Informazioni aggiuntive 575
Esegui la migrazione di uno stack ELK nel cloud AWS 580
Riepilogo 580
Prerequisiti e limitazioni 580
Architecture 582
Tools (Strumenti) 584
Epiche 585
Risorse correlate 592
Informazioni aggiuntive 594
Migra i dati verso Starburst AWS 595
Riepilogo 595

Prerequisiti e limitazioni 595
Architecture 595
Tools (Strumenti) 597
Epiche 598
Risorse correlate 601
Ottimizza l'ingestione ETL delle dimensioni del file di input 602
Riepilogo 602
Prerequisiti e limitazioni 602
Architecture 602
Tools (Strumenti) 603
Epiche 603
Risorse correlate 606
Informazioni aggiuntive 607
Orchestra una pipeline ETL con Step Functions 608
Riepilogo 608
Prerequisiti e limitazioni 608
Architecture 609
Tools (Strumenti) 610
Epiche 611
risoluzione dei problemi 618
Risorse correlate 619
Informazioni aggiuntive 619
Esegui analisi ML utilizzando Amazon Redshift ML 620
Riepilogo 620
Prerequisiti e limitazioni 620
Architecture 621
Tools (Strumenti) 622
Epiche 623
Risorse correlate 626
Interroga Amazon DynamoDB utilizzando Amazon Athena 628
Riepilogo 628
Prerequisiti e limitazioni 628
Architecture 629
Tools (Strumenti) 629
Epiche 630
risoluzione dei problemi 633

Risorse correlate 634
Interroga le tabelle DynamoDB usando Athena 635
Riepilogo 635
Prerequisiti e limitazioni 635
Architecture 636
Tools (Strumenti) 636
Epiche 637
Risorse correlate 645
Informazioni aggiuntive 645
Imposta uno spazio dati minimo utilizzabile 647
Riepilogo 647
Prerequisiti e limitazioni 648
Architecture 649
Tools (Strumenti) 650
Best practice 651
Epiche 651
Risoluzione dei problemi 706
Risorse correlate 706
Informazioni aggiuntive 706
Imposta l'ordinamento specifico per lingua per i risultati delle query di Amazon Redshift 712
Riepilogo 712
Prerequisiti e limitazioni 712
Tools (Strumenti) 712
Epiche 713
Risorse correlate 718
Informazioni aggiuntive 718
Sottoscrivi una funzione Lambda alle notifiche di eventi dai bucket S3 interregionali 722
Riepilogo 722
Prerequisiti e limitazioni 722
Architecture 723
Tools (Strumenti) 723
Epiche 724
Risorse correlate 728
Tre tipi di job AWS Glue per la conversione dei dati 730
Riepilogo 730
Prerequisiti e limitazioni 730

Architecture ... 730
Tools (Strumenti) ... 731
Epiche ... 732
Risorse correlate ... 734
Informazioni aggiuntive ... 735
Allegati ... 741
Visualizza i log di controllo di Amazon Redshift utilizzando Athena e QuickSight ... 742
Riepilogo ... 742
Prerequisiti e limitazioni ... 742
Architecture ... 742
Tools (Strumenti) ... 743
Epiche ... 743
Risorse correlate ... 748
Allegati ... 748
Visualizza i report sulle credenziali IAM utilizzando Amazon Quick Sight ... 749
Riepilogo ... 749
Prerequisiti e limitazioni ... 749
Architecture ... 750
Tools (Strumenti) ... 751
Epiche ... 751
Informazioni aggiuntive ... 757
Altri modelli ... 759
Calcolo ... 761
Contenitori e microservizi ... 762
Accedi ad Amazon Neptune da Amazon EKS ... 764
Accedi alle applicazioni container su Amazon ECS ... 776
Accedi alle applicazioni container su Amazon ECS con un tipo di lancio AWS Fargate ... 793
Accedi alle applicazioni container in modo privato su Amazon EKS ... 808
Automatizza i backup per le istanze DB di Amazon RDS for PostgreSQL ... 816
Automatizza la distribuzione di Node Termination Handler ... 829
Crea e distribuisci automaticamente un'applicazione Java su Amazon EKS ... 844
Copia le immagini dei container Amazon ECR tra account e regioni ... 869
Crea una definizione di attività Amazon ECS sulle EC2 istanze utilizzando Amazon EFS ... 896
Implementa le funzioni Lambda con immagini dei container ... 902
Distribuisci microservizi Java su Amazon ECS utilizzando AWS Fargate ... 911
Distribuisci pacchetti Kubernetes utilizzando Amazon EKS e Helm ... 917

Implementa un cluster CockroachDB in Amazon EKS utilizzando Terraform 928
Implementa un microservizio Java su Amazon EKS ed esponilo con un Application Load Balancer 939
Implementa un'applicazione basata su gRPC su Amazon EKS 953
Distribuisci applicazioni containerizzate su AWS IoT Greengrass V2 965
Distribuisci contenitori utilizzando Elastic Beanstalk 990
Genera un indirizzo IP statico in uscita utilizzando Lambda e Amazon VPC 996
Identifica automaticamente le immagini duplicate dei container 1010
Installa l'agente SSM sui nodi di lavoro Amazon EKS 1039
Installa l'agente SSM e l' CloudWatch agente sui nodi di lavoro Amazon EKS utilizzando preBootstrapCommands 1045
Migra i controller di ingresso NGINX per la modalità automatica EKS 1052
Esegui la migrazione dei carichi di lavoro dei container da ARO a ROSA 1073
Esegui attività Amazon ECS su Amazon WorkSpaces 1089
Esegui un contenitore Docker per API Web ASP.NET su AWS 1099
Esegui carichi di lavoro con stato con archiviazione persistente dei dati 1111
Configura la scalabilità automatica utilizzando Amazon EKS Pod Identity e KEDA 1137
..... 1160
Usa il TLS reciproco in Amazon ECS con ALB 1183
Altri modelli 1214
Serverless 1216
Crea un'app React Native con AWS Amplify 1217
Gestisci centralmente i tenant su più prodotti SaaS 1236
Consolida la generazione di URL predefiniti di Amazon S3 e il download di oggetti 1249
Crea una connessione Amazon EventBridge tra più account 1265
Distribuisci record DynamoDB ad Amazon S3 1280
Implementa il controllo delle versioni delle API basato sul percorso utilizzando API Gateway 1287
Importa la libreria psycopg2 in AWS Lambda 1299
Integrazione di API Gateway con Amazon SQS 1308
Elabora in modo APIs asincrono con AWS Lambda 1324
Elabora in modo APIs asincrono con Amazon DynamoDB Streams 1333
Elabora in modo APIs asincrono con Amazon SQS 1344
Esegui le attività di Systems Manager Automation in modo sincrono da Step Functions 1355
Esegui letture parallele di oggetti S3 con AWS Lambda 1369
Invia dati di telemetria da Lambda a OpenSearch 1381

Configura un router cellulare senza server ... 1397
Configurare l'accesso privato a un bucket Amazon S3 ... 1416
Risoluzione dei problemi relativi agli stati di Step Functions ... 1423
Altri modelli ... 1433
Rete ... 1435
Automatizza il peering per AWS Transit Gateway ... 1436
Centralizza la connettività di rete utilizzando AWS Transit Gateway ... 1443
Configurare la crittografia HTTPS per Oracle JD Edwards EnterpriseOne utilizzando un Application Load Balancer ... 1450
Connect ai dati e ai piani di controllo dell'Application Migration Service tramite una rete privata ... 1461
Crea oggetti Infoblox utilizzando risorse personalizzate AWS CloudFormation ... 1475
Crea un'architettura IPAM multiregionale utilizzando Terraform ... 1488
Personalizza CloudWatch gli avvisi per Network Firewall ... 1504
Distribuisci risorse in una Wavelength Zone usando Terraform ... 1523
Migrazione di record DNS in blocco su una zona ospitata privata Route 53 ... 1531
Modifica le intestazioni HTTP durante la migrazione da F5 a un Application Load Balancer su AWS ... 1540
Riporta i risultati di Network Access Analyzer in più Account AWS ... 1546
Configura la risoluzione DNS per reti ibride in un ambiente con più account AWS ... 1576
... 1591
Visualizza i log e i parametri di AWS Network Firewall utilizzando Splunk ... 1597
Altri modelli ... 1608
Distribuzione di contenuti ... 1609
Inviare AWS WAF i log a Splunk utilizzando Firehose ... 1610
Offri contenuti statici in un bucket S3 tramite un VPC utilizzando CloudFront ... 1618
Altri modelli ... 1627
Database e archiviazione ... 1628
Database ... 1629
Accedi ai dati di SQL Server locali utilizzando server collegati ... 1632
Aggiungi HA a Oracle PeopleSoft su AWS ... 1638
Analizza gli oggetti per le migrazioni parziali da Oracle a PostgreSQL ... 1663
Valuta le prestazioni delle query per migrare SQL Server a MongoDB ... 1680
Automatizza le blue/green implementazioni dei database globali Aurora con IaC ... 1690
Automatizza la replica tra account delle istanze Amazon RDS ... 1723
Automatizza le attività del database con Task Scheduler AWS Lambda ... 1737

Automatizza il failover e il failback con DR Orchestrator Framework 1749
Esegui automaticamente il backup dei database SAP HANA 1776
Genera automaticamente modelli PynamoDB e funzioni CRUD per DynamoDB 1784
Blocca l'accesso pubblico ad Amazon RDS 1792
Configurazione dell'autenticazione Windows per Amazon RDS utilizzando Microsoft AD ... 1799
Configurazione dell'accesso ad Amazon DynamoDB su più account 1815
Configura il routing di sola lettura in un gruppo di disponibilità Always On 1828
Connect utilizzando un tunnel SSH in pGAdmin 1837
Convertire le query JSON Oracle in SQL del database PostgreSQL 1842
Copia le tabelle Amazon DynamoDB tra più account 1872
Copia le tabelle Amazon DynamoDB tra più account 1879
Crea report su costi e utilizzo per Amazon RDS e Amazon Aurora 1892
Implementa SQL Server FCIs su Amazon EC2 e Amazon FSx 1899
Emula gli array PL/SQL associativi Oracle in Amazon Aurora e Amazon RDS 1915
Abilita connessioni crittografate per le istanze DB PostgreSQL 1927
Crittografa un'istanza database Amazon RDS for PostgreSQL esistente 1936
Applica il tagging automatico dei database Amazon RDS al momento del lancio 1944
Stima dei costi di DynamoDB 1950
Stima dei costi di storage per una tabella Amazon DynamoDB 1961
Stima delle dimensioni del motore Amazon RDS per un database Oracle utilizzando i report AWR 1965
Esportazione di tabelle Amazon RDS for SQL Server in un bucket S3 1997
Gestisci i blocchi anonimi nelle istruzioni SQL dinamiche 2008
Aiutaci a far rispettare il tagging di DynamoDB 2016
Implementa il disaster recovery interregionale 2022
Esegui la migrazione di oltre 100 argomenti di funzioni Oracle a PostgreSQL 2040
Esegui la migrazione dei carichi di lavoro Redis su Redis Enterprise Cloud su AWS 2046
Migrazione di SAP HANA su AWS utilizzando HSR 2073
Esegui la migrazione del gruppo di disponibilità SQL Server Always On utilizzando MGN AWS 2086
Esegui la migrazione di SQL Server su AWS utilizzando gruppi di disponibilità distribuiti ... 2098
Migrare un database relazionale su MongoDB Atlas su AWS 2110
Migrare un ambiente MongoDB ospitato autonomamente su MongoDB Atlas 2120
Migrazione di un database Oracle su Amazon DynamoDB 2132
Esegui la migrazione da Oracle 8i o 9i ad Amazon RDS for Oracle utilizzando AWS DMS SharePlex 2137

Esegui la migrazione da MySQL locale ad Amazon EC2 2146
Monitora Amazon Aurora per la crittografia 2155
Monitora GoldenGate i log utilizzando Amazon CloudWatch 2161
Ripiattaforma Oracle Database EE su Amazon RDS per Oracle SE2 2174
Replica i database mainframe su AWS utilizzando Precisly Connect 2185
Pianifica i lavori per Amazon RDS e Aurora PostgreSQL 2204
Invia notifiche per RDS for SQL Server utilizzando un server SMTP 2210
Configura DR per SAP su IBM Db2 su AWS 2224
Configura una CI/CD pipeline per la migrazione del database utilizzando Terraform 2246
Configura un Microsoft SQL Server FCI su Amazon utilizzando EC2 FSx 2255
Configurare un' HA/DR architettura per Oracle E-Business Suite su Amazon RDS Custom 2274
Trasmetti dati dai database a MongoDB Atlas su AWS 2283
Ruoli di transizione per un'applicazione Oracle PeopleSoft 2294
Scarica dati da Amazon Redshift su più account 2330
Modelli di migrazione del database per carico di lavoro 2359
Altri modelli 2370
Archiviazione e backup 2375
Consenti alle EC2 istanze l'accesso in scrittura ai bucket S3 in AMS 2376
Automatizza l'inserimento del flusso di dati in un database Snowflake 2382
Crittografa automaticamente i volumi EBS 2393
Effettua il backup dei server Sun SPARC nell'emulatore Charon-SSP su AWS 2405
Esegui il backup e l'archiviazione dei dati su Amazon S3 con Veeam 2429
Copia i dati di Amazon S3 tra account e regioni 2455
Abilita l'archiviazione dei DB2 log su Amazon S3 2470
Esegui la migrazione dei dati Hadoop su Amazon S3 utilizzando DistCp e AWS per Amazon S3 PrivateLink 2487
Altri modelli 2503
Strumenti per sviluppatori 2505
DevOps 2506
Accelera MLOps con Backstage e modelli Amazon SageMaker AI self-service 2509
Automatizza le operazioni AWS dell'infrastruttura 2529
Automatizza CloudFront gli aggiornamenti per le modifiche negli endpoint del bilanciamento del carico 2538
Automatizza le CodeGuru revisioni per le applicazioni AWS CDK Python 2547
Automatizza Catena di approvvigionamento di AWS l'implementazione dei data lake 2559
Automatizza la valutazione delle risorse AWS 2595

Automatizza l'installazione dei sistemi SAP 2609
Automatizza la distribuzione del portafoglio e dei prodotti di Service Catalog utilizzando AWS CDK 2622
Automatizza una pipeline dinamica di hotfix 2641
Automatizza l'eliminazione degli stack CloudFormation 2671
Automatizza l'inserimento di metriche personalizzate di Amazon MWAA 2683
Automatizza la distribuzione di stack set utilizzando AWS e AWS CodePipeline CodeBuild 2700
Allega automaticamente una policy gestita per Systems Manager ai profili delle EC2 istanze 2730
Crea automaticamente CI/CD pipeline e cluster Amazon ECS per microservizi 2746
Crea e invia immagini Docker ad Amazon ECR 2759
Crea e testa app iOS con i servizi AWS 2765
Configurazione del TLS reciproco per le applicazioni su Amazon EKS 2771
Crea risorse Amazon WorkSpaces Applications utilizzando CloudFormation 2783
Crea un parser di log personalizzato per Amazon ECS utilizzando Firelens 2790
Crea un framework di orchestrazione delle risorse 2800
Crea richieste pull automatizzate per l'infrastruttura AWS 2813
Crea pipeline CI dinamiche per progetti Java e Python 2825
Implementa i canarini CloudWatch Synthetics 2842
Implementa una ChatOps soluzione per gestire i risultati delle scansioni SAST 2849
Implementa sistemi agentici su Amazon Bedrock con CrewAI 2867
..... 2885
Distribuisci codice in più regioni AWS 2889
Distribuisci carichi di lavoro dalle DevOps pipeline di Azure ai cluster Amazon EKS privati 2901
Esegui query SQL su Amazon Redshift 2935
Esporta i report di AWS Backup come file CSV 2948
Esporta i tag delle EC2 istanze Amazon in un file CSV 2958
Genera un CloudFormation modello AWS contenente le regole gestite di AWS Config 2965
Concedi alle istanze di SageMaker notebook l'accesso a un repository da più account CodeCommit 2972
Implementa una strategia di ramificazione GitHub Flow 2983
Implementa una strategia di ramificazione Gitflow 2992
Implementa una strategia di ramificazione Trunk 3004
Implementa la scansione Checkov personalizzata centralizzata 3010
Implementa la diagnostica con K8sgPT e Amazon Bedrock 3021
Avvia diverse CI/CD pipeline dopo aver rilevato le modifiche in un monorepo 3035

Integra un repository Bitbucket con AWS Amplify 3053
Avvia un CodeBuild progetto su più account AWS usando Lambda 3061
Gestione del failover Multi-AZ per i cluster EMR utilizzando Application Recovery Controller 3074
Gestisci le blue/green distribuzioni di microservizi su più account e regioni 3089
Monitora i repository Amazon ECR per le autorizzazioni wildcard 3123
Ottimizza le implementazioni senza server multi-account 3128
Fornisci AWS Service Catalog prodotti utilizzando GitHub Actions 3149
Fornisci ruoli IAM con privilegi minimi 3159
Pubblica i CloudWatch parametri di Amazon in un file CSV 3172
Rimuovi le EC2 voci Amazon ovunque Account AWS 3178
Rimuovi le EC2 voci Amazon nello stesso Account AWS 3196
Esegui test unitari per i lavori ETL in Python in AWS Glue 3207
Configura una pipeline con CI/CD CodePipeline 3218
Configura la registrazione centralizzata su larga scala utilizzando Terraform 3234
end-to-endConfigurazione della crittografia per le applicazioni su Amazon EKS 3260
Semplifica la distribuzione di applicazioni multi-tenant Amazon EKS 3274
Semplifica i flussi di lavoro Amazon Lex 3293
Usa il costrutto hook AWS Fargate WaitCondition 3313
Usa repository Git di terze parti in AWS CodePipeline 3326
Convalida le configurazioni Terraform utilizzando AWS CodePipeline 3336
Altri modelli 3352
Infrastruttura 3356
Accedi a un host bastion utilizzando Session Manager e Instance Connect 3358
Centralizza la risoluzione DNS utilizzando AWS Managed Microsoft AD 3374
Centralizza il monitoraggio utilizzando Observability Access Manager 3386
Verifica la presenza di tag obbligatori EC2 nelle istanze al momento del lancio 3400
Pulisci le risorse AFT in modo sicuro dopo la perdita dei file statali 3406
Crea una pipeline nelle regioni AWS che non supportano AWS CodePipeline 3419
Personalizza i nomi dei ruoli predefiniti utilizzando AWS CDK gli aspetti e gli escape hatch 3427
Implementa un cluster Cassandra su Amazon EC2 con dati statici privati IPs 3436
Estendi VRFs ad AWS utilizzando Transit Gateway Connect 3443
Ricevi notifiche Amazon SNS per le modifiche di stato alle chiavi AWS KMS 3461
Conserva lo spazio IP instradabile nei progetti VPC multi-account per sottoreti non destinate ai carichi di lavoro 3468
Fornisci un prodotto Terraform in Service Catalog da un repository 3474

Registrane più di uno Account AWS con un unico indirizzo e-mail 3493
Configura la risoluzione DNS per reti ibride in un ambiente AWS con account singolo 3511
Configura automaticamente i bot UiPath RPA su Amazon EC2 3516
Configura un' PeopleSoft architettura ad alta disponibilità su AWS 3533
Configura il disaster recovery per Oracle JD Edwards EnterpriseOne 3562
Imposta il rilevamento e la segnalazione della deriva 3590
Importa con successo un bucket S3 come stack CloudFormation 3595
Sincronizza i file system Amazon EFS in diverse regioni 3607
AWS Infrastruttura di test utilizzando LocalStack Terraform Tests 3615
Aggiorna i cluster SAP Pacemaker da a ENSA1 ENSA2 3623
Usa zone di disponibilità coerenti VPCs in diversi account 3645
Usa l'utente IDs nelle politiche IAM 3651
Convalida il codice Account Factory for Terraform localmente 3662
Altri modelli 3682
App Web e mobili 3686
Autentica gli utenti dell'applicazione web React 3687
Crea un'app React utilizzando AWS Amplify e Amazon Cognito 3697
Crea un portale per micro-frontend 3713
Implementa una SPA basata su React su Amazon S3 e CloudFront 3744
Implementa un'API Amazon API Gateway utilizzando endpoint privati 3752
Incorpora i componenti visivi di Amazon Quick Sight nelle applicazioni Web 3759
Esplora lo sviluppo di app web con Green Boost 3781
Esegui test unitari utilizzando AWS CodeBuild 3809
Struttura un progetto Python in architettura esagonale 3817
Altri modelli 3844
IoT 3846
Estrai e interroga gli attributi AWS IoT SiteWise dei metadati 3847
Riepilogo 3847
Prerequisiti e limitazioni 3847
Architecture 3848
Tools (Strumenti) 3848
Epiche 3849
Risorse correlate 3852
Informazioni aggiuntive 3852
..... 3855
Riepilogo 3855

Prerequisiti e limitazioni ... 3856
Architecture ... 3856
Tools (Strumenti) ... 3857
Best practice ... 3857
Epiche ... 3858
risoluzione dei problemi ... 3873
Risorse correlate ... 3875
Informazioni aggiuntive ... 3875
Altri modelli ... 3877
Migrazione e modernizzazione ... 3878
Migrazione ... 3879
Crea CloudFormation modelli AWS per AWS DMS ... 3880
Inizia con l'individuazione automatica dei portafogli ... 3884
Migra i carichi di lavoro Cloudera locali su AWS ... 3892
Risolvi gli errori di connessione dopo la migrazione di SQL Server su AWS ... 3908
Riavvia automaticamente AWS Replication Agent senza disabilitarlo SELinux ... 3911
Re-architetto ... 3919
Riospitare ... 4298
Trasferisci ... 4532
Conversione piattaforma ... 4575
Modelli di migrazione per carico di lavoro ... 5009
Altri modelli ... 5018
Modernizzazione ... 5020
Archivia automaticamente i dati DynamoDB scaduti su Amazon S3 ... 5021
Crea un'architettura serverless multi-tenant in Amazon Service OpenSearch ... 5038
Distribuisci applicazioni a stack multiplo ... 5087
Distribuisci applicazioni annidate utilizzando AWS SAM ... 5096
Implementa l'isolamento dei tenant SaaS per Amazon S3 utilizzando una TVM AWS Lambda ... 5105
Implementa il modello di saga serverless usando Step Functions ... 5130
Gestisci le applicazioni container locali con Amazon ECS Anywhere ... 5145
Modernizza le applicazioni ASP.NET Web Forms su AWS ... 5155
Esegui carichi di lavoro basati su eventi con AWS Fargate ... 5171
Onboarding dei tenant nell'architettura SaaS ... 5181
Usa CQRS e l'event sourcing ... 5206
Altri modelli ... 5231

Mainframe 5234
Accesso Servizi AWS da IBM z/OS installando il AWS CLI 5236
Esegui il backup e l'archiviazione dei dati del mainframe su Amazon S3 5256
Crea programmi COBOL Db2 usando e Modernizzazione del mainframe AWS AWS CodeBuild 5284
Create un Micro Focus Enterprise Server PAC 5308
Crea un visualizzatore di file mainframe nel cloud AWS 5328
Containerizza le applicazioni Blu Age modernizzate 5344
Convertire i dati EBCDIC in ASCII su AWS 5354
Converti file EBCDIC mainframe in file ASCII utilizzando AWS Lambda 5373
Convertite i file di dati mainframe con layout di record complessi 5396
Implementa un ambiente per app containerizzate 5414
Genera approfondimenti sui dati Db2 z/OS 5424
Genera approfondimenti utilizzando Modernizzazione del mainframe AWS Amazon Q 5476
Implementazione dell'autenticazione Microsoft Entra ID per AWS Blu Age 5494
Integra il controller universale Stonebranch con AWS 5513
Migra e replica i file VSAM nel cloud AWS utilizzando Precisly 5547
Modernizza un'applicazione mainframe utilizzando AWS Transform 5565
Modernizza le applicazioni mainframe utilizzando AWS Transform 5584
Modernizza la gestione dell'output del mainframe su AWS 5620
Modernizza i tuoi carichi di lavoro di stampa in batch su mainframe su AWS 5668
Modernizza il tuo ambiente mainframe con Rocket Software Enterprise Suite 5698
Modernizza i tuoi carichi di lavoro di stampa online mainframe su AWS 5715
Sposta i file mainframe su Amazon S3 utilizzando Transfer Family 5750
Ottimizza le prestazioni della tua applicazione modernizzata AWS Blu Age 5763
Accesso sicuro degli utenti in un database federativo Db2 5788
Trasferimento di z/OS dati Db2 su AWS 5797
Altri modelli 5825
Gestione 5826
Gestione dei costi 5827
Crea report dettagliati su costi e utilizzo per i lavori AWS Glue 5828
Crea report dettagliati su costi e utilizzo per i cluster Amazon EMR 5833
Altri modelli 5837
Informatica per l'utente finale 5838
Implementa l'autenticazione SAML 2.0 per Amazon WorkSpaces 5839
Altri modelli 5849

High Performance Computing 5850
Implementa un file system Lustre utilizzando Terraform e DRA 5851
Configura una dashboard di monitoraggio Grafana per AWS ParallelCluster 5859
Altri modelli 5873
Cloud ibrido 5874
Configura una pipeline per il carico di lavoro ibrido su Amazon ECS Anywhere 5875
Altri modelli 5895
Gestione e governance 5896
Avvisa quando le risorse Amazon Data Firehose non sono crittografate 5897
Automatizza le allocazioni IPAM CIDR di Amazon VPC IPv4 5901
Automatizza l'aggiunta o l'aggiornamento delle voci di registro di Windows 5932
Crea automaticamente un RFC usando Python 5937
Arresta e avvia automaticamente un'istanza database Amazon RDS 5944
Centralizza la distribuzione dei pacchetti software in AWS Organizations utilizzando Terraform 5959
Configura la registrazione per le applicazioni.NET in Logs CloudWatch 5972
Copia i prodotti AWS Service Catalog tra account e regioni AWS 5982
Crea una matrice RACI per le operazioni cloud 5992
Crea allarmi per metriche personalizzate utilizzando CloudWatch 5998
Crea un IDE AWS Cloud9 con volumi EBS crittografati predefiniti 6006
Crea dashboard basate su tag automaticamente CloudWatch 6012
Documenta il design della tua landing zone 6021
Abilita Amazon DevOps Guru in tutta l'organizzazione con AWS CDK 6025
Gestisci i set di autorizzazioni per più account utilizzando AFT 6051
Implementa AFT utilizzando una pipeline di bootstrap 6063
Gestisci i prodotti AWS Service Catalog in più account e regioni AWS 6088
Monitora i cluster SAP RHEL Pacemaker 6096
Monitora l'attività delle applicazioni utilizzando CloudWatch Logs Insights 6115
Monitora l'uso di un'AMI su Account AWS 6127
Visualizza i dettagli degli snapshot EBS per il tuo account o la tua organizzazione AWS ... 6143
Altri modelli 6150
Messaggi e comunicazioni 6153
Automatizza la configurazione di RabbitMQ in Amazon MQ 6154
Migliora la qualità delle chiamate sulle postazioni di lavoro degli agenti in Amazon Connect 6162
Altri modelli 6178

Strategia multi-account 6179
Migra un uomo da a Account AWS AWS Organizations AWS Control Tower 6180
Imposta avvisi per la chiusura programmatica degli account in AWS Organizations 6190
Altri modelli 6200
Sicurezza, identità e conformità 6201
Automatizza la risposta agli incidenti e l'analisi forense 6203
Riepilogo 6203
Prerequisiti e limitazioni 6203
Architecture 6204
Tools (Strumenti) 6206
Epiche 6207
Risorse correlate 6211
Informazioni aggiuntive 6211
Allegati 6212
Controlla automaticamente l'accesso da indirizzi IP pubblici 6213
Riepilogo 6213
Prerequisiti e limitazioni 6213
Architecture 6214
Tools (Strumenti) 6216
Best practice 6217
Epiche 6217
risoluzione dei problemi 6221
Risorse correlate 6222
Automatically remediate unencrypted Amazon RDS DB instances and clusters 6223
Riepilogo 6223
Prerequisiti e limitazioni 6223
Architecture 6225
Tools (Strumenti) 6225
Best practice 6226
Epiche 6227
Risorse correlate 6233
Informazioni aggiuntive 6233
Convalida e implementa automaticamente le politiche e i ruoli IAM 6235
Riepilogo 6235
Prerequisiti e limitazioni 6236
Architecture 6236

Tools (Strumenti) 6237
Epiche 6238
Risorse correlate 6242
Integrazione bidirezionale di Security Hub, CSPM e Jira 6243
Riepilogo 6243
Prerequisiti e limitazioni 6244
Architecture 6245
Tools (Strumenti) 6246
Epiche 6247
Risorse correlate 6257
Informazioni aggiuntive 6257
Crea una pipeline per immagini di contenitori rinforzati 6259
Riepilogo 6259
Prerequisiti e limitazioni 6259
Architecture 6260
Tools (Strumenti) 6263
Epiche 6263
risoluzione dei problemi 6272
Risorse correlate 6272
Centralizza la gestione delle chiavi di accesso IAM in AWS Organizations utilizzando Terraform 6274
Riepilogo 6274
Prerequisiti e limitazioni 6275
Architecture 6275
Tools (Strumenti) 6277
Best practice 6278
Poemi epici 6278
risoluzione dei problemi 6287
Risorse correlate 6288
Controlla una CloudFront distribuzione Amazon per la registrazione degli accessi, la versione HTTPS e TLS 6289
Riepilogo 6289
Prerequisiti e limitazioni 6289
Architecture 6290
Tools (Strumenti) 6290
Epiche 6291

Risorse correlate 6294
Allegati 6294
Scegli un flusso di autenticazione Amazon Cognito 6295
Riepilogo 6295
Prerequisiti e limitazioni 6295
Architecture 6296
Tools (Strumenti) 6300
Epiche 6301
Risorse correlate 6304
Informazioni aggiuntive 6305
Crea regole AWS Config personalizzate usando Guard 6306
Riepilogo 6306
Prerequisiti e limitazioni 6307
Architecture 6307
Tools (Strumenti) 6312
Epiche 6312
risoluzione dei problemi 6314
Risorse correlate 6315
Crea un rapporto sui risultati di Prowler tra più Account AWS 6316
Riepilogo 6316
Prerequisiti e limitazioni 6317
Architecture 6318
Tools (Strumenti) 6318
Epiche 6320
risoluzione dei problemi 6344
Risorse correlate 6345
Informazioni aggiuntive 6345
Implementa i controlli AWS Control Tower utilizzando AWS CDK 6347
Riepilogo 6347
Prerequisiti e limitazioni 6348
Architecture 6349
Tools (Strumenti) 6350
Best practice 6351
Epiche 6351
Risorse correlate 6358
Informazioni aggiuntive 6359

Implementa i controlli utilizzando AWS Control Tower Terraform 6362
Riepilogo 6362
Prerequisiti e limitazioni 6363
Architecture 6364
Tools (Strumenti) 6364
Best practice 6365
Epiche 6365
risoluzione dei problemi 6371
Risorse correlate 6373
Informazioni aggiuntive 6373
Implementa le automazioni di sicurezza per l'utilizzo di Terraform AWS WAF 6377
Riepilogo 6377
Prerequisiti e limitazioni 6377
Architecture 6378
Tools (Strumenti) 6378
Best practice 6379
Epiche 6379
risoluzione dei problemi 6383
Risorse correlate 6383
Implementa una pipeline che rilevi i problemi di sicurezza nel codice 6384
Riepilogo 6384
Prerequisiti e limitazioni 6384
Architecture 6384
Tools (Strumenti) 6385
Epiche 6386
risoluzione dei problemi 6389
Risorse correlate 6389
Informazioni aggiuntive 6389
Implementa controlli investigativi per sottoreti pubbliche 6392
Riepilogo 6392
Prerequisiti e limitazioni 6393
Architecture 6393
Tools (Strumenti) 6394
Best practice 6395
Epiche 6395
Risorse correlate 6403

Informazioni aggiuntive 6404
Implementa controlli preventivi per le sottoreti pubbliche 6406
Riepilogo 6406
Prerequisiti e limitazioni 6406
Architecture 6407
Tools (Strumenti) 6408
Epiche 6409
Risorse correlate 6415
Informazioni aggiuntive 6415
Rileva le istanze Amazon RDS con certificati CA in scadenza 6418
Riepilogo 6418
Prerequisiti e limitazioni 6418
Architecture 6419
Tools (Strumenti) 6421
Best practice 6421
Epiche 6422
risoluzione dei problemi 6426
Risorse correlate 6427
Genera dinamicamente una policy IAM con IAM Access Analyzer 6428
Riepilogo 6428
Prerequisiti e limitazioni 6428
Architecture 6429
Tools (Strumenti) 6430
Epiche 6431
Risorse correlate 6438
Abilita l' GuardDuty utilizzo di modelli CloudFormation 6440
Riepilogo 6440
Prerequisiti e limitazioni 6440
Architecture 6441
Tools (Strumenti) 6441
Epiche 6442
Risorse correlate 6446
Informazioni aggiuntive 6447
Abilita la crittografia trasparente dei dati in Amazon RDS for SQL Server 6448
Riepilogo 6448
Prerequisiti e limitazioni 6448

Architecture 6448
Tools (Strumenti) 6449
Epiche 6449
Risorse correlate 6452
Aiuta a prevenire l'eliminazione pianificata delle chiavi KMS 6453
Riepilogo 6453
Prerequisiti e limitazioni 6453
Architecture 6454
Tools (Strumenti) 6455
Epiche 6456
Risorse correlate 6459
Informazioni aggiuntive 6460
Allegati 6460
Identifica i bucket Amazon S3 pubblici in AWS Organizations 6461
Riepilogo 6461
Prerequisiti e limitazioni 6461
Architecture 6462
Tools (Strumenti) 6463
Epiche 6464
risoluzione dei problemi 6468
Risorse correlate 6469
Informazioni aggiuntive 6469
Inserisci i log di sicurezza di AWS in Microsoft Sentinel 6470
Riepilogo 6470
Prerequisiti e limitazioni 6470
Architecture 6472
Tools (Strumenti) 6473
Best practice 6474
Epiche 6474
Risorse correlate 6484
Gestisci AWS Organizations le politiche come codice 6486
Riepilogo 6486
Prerequisiti e limitazioni 6486
Architecture 6487
Tools (Strumenti) 6488
Best practice 6490

Epiche 6490
risoluzione dei problemi 6508
Risorse correlate 6510
Informazioni aggiuntive 6511
Gestisci i set di autorizzazioni di IAM Identity Center utilizzando CodePipeline 6513
Riepilogo 6513
Prerequisiti e limitazioni 6513
Architecture 6515
Tools (Strumenti) 6516
Best practice 6517
Epiche 6517
risoluzione dei problemi 6527
Risorse correlate 6527
Gestisci le credenziali con AWS Secrets Manager 6528
Riepilogo 6528
Prerequisiti e limitazioni 6528
Architecture 6528
Tools (Strumenti) 6529
Epiche 6529
Risorse correlate 6531
Informazioni aggiuntive 6531
Monitora ElastiCache i cluster Amazon per la crittografia a riposo 6534
Riepilogo 6534
Prerequisiti e limitazioni 6535
Architecture 6535
Tools (Strumenti) 6536
Best practice 6537
Epiche 6538
risoluzione dei problemi 6541
Risorse correlate 6542
Monitoraggio dell'attività dell'utente root IAM 6543
Riepilogo 6543
Prerequisiti e limitazioni 6543
Architecture 6544
Tools (Strumenti) 6544
Epiche 6546

Risorse correlate 6551
Informazioni aggiuntive 6551
Notifica quando viene creato un utente IAM 6552
Riepilogo 6552
Prerequisiti e limitazioni 6552
Architecture 6552
Tools (Strumenti) 6553
Epiche 6554
Risorse correlate 6556
Allegati 6556
Impedisci l'accesso a Internet utilizzando un SCP 6557
Riepilogo 6557
Prerequisiti e limitazioni 6557
Tools (Strumenti) 6558
Best practice 6558
Epiche 6559
Risorse correlate 6561
Segnala le identità di IAM Identity Center e le relative assegnazioni 6562
Riepilogo 6562
Prerequisiti e limitazioni 6562
Architecture 6564
Tools (Strumenti) 6564
Epiche 6564
risoluzione dei problemi 6566
Risorse correlate 6568
Informazioni aggiuntive 6568
Limita l'accesso in base all'indirizzo IP o alla geolocalizzazione 6571
Riepilogo 6571
Prerequisiti e limitazioni 6571
Tools (Strumenti) 6572
Epiche 6573
Risorse correlate 6578
Scansiona gli archivi Git alla ricerca di informazioni sensibili 6580
Riepilogo 6580
Prerequisiti e limitazioni 6580
Architecture 6580

Tools (Strumenti) 6580
Best practice 6581
Epiche 6581
Risorse correlate 6586
Trasferimenti sicuri di file con Transfer Family e GuardDuty 6587
Riepilogo 6587
Prerequisiti e limitazioni 6588
Architecture 6588
Tools (Strumenti) 6590
Best practice 6591
Epiche 6591
risoluzione dei problemi 6597
Risorse correlate 6600
Informazioni aggiuntive 6600
Proteggi i dati sensibili nei CloudWatch registri con Macie 6602
Riepilogo 6602
Prerequisiti e limitazioni 6602
Architecture 6603
Tools (Strumenti) 6604
Best practice 6605
Epiche 6605
risoluzione dei problemi 6626
Risorse correlate 6626
Invia avvisi da AWS Network Firewall a un canale Slack 6628
Riepilogo 6628
Prerequisiti e limitazioni 6628
Architecture 6629
Tools (Strumenti) 6630
Epiche 6631
Risorse correlate 6637
Informazioni aggiuntive 6637
Invia attributi personalizzati ai token Amazon Cognito 6642
Riepilogo 6642
Prerequisiti e limitazioni 6642
Architecture 6643
Tools (Strumenti) 6644

Best practice 6644
Epiche 6646
risoluzione dei problemi 6655
Risorse correlate 6655
Informazioni aggiuntive 6655
Semplifica la gestione privata dei certificati utilizzando AWS Private CA e AWS RAM 6659
Riepilogo 6659
Prerequisiti e limitazioni 6659
Architecture 6660
Tools (Strumenti) 6661
Epiche 6662
Risorse correlate 6670
Informazioni aggiuntive 6670
Semplifica la gestione EC2 della conformità di Amazon con agenti Amazon Bedrock e AWS Config 6671
Riepilogo 6671
Prerequisiti e limitazioni 6671
Architecture 6672
Tools (Strumenti) 6673
Best practice 6674
Epiche 6674
risoluzione dei problemi 6677
Risorse correlate 6677
Aggiorna le credenziali da IAM Identity Center utilizzando PowerShell 6678
Riepilogo 6678
Prerequisiti e limitazioni 6678
Architecture 6679
Tools (Strumenti) 6679
Best practice 6680
Epiche 6680
risoluzione dei problemi 6682
Risorse correlate 6683
Informazioni aggiuntive 6683
Usa Network Firewall per acquisire i nomi DNS dal traffico in uscita 6686
Riepilogo 6686
Prerequisiti e limitazioni 6686

Architecture 6687
Tools (Strumenti) 6687
Epiche 6688
Usa Terraform per abilitare automaticamente GuardDuty 6704
Riepilogo 6704
Prerequisiti e limitazioni 6705
Architecture 6707
Tools (Strumenti) 6708
Epiche 6709
Risorse correlate 6718
Informazioni aggiuntive 6719
Verifica le best practice per PCI DSS 4.0 6720
Riepilogo 6720
Prerequisiti e limitazioni 6720
Tools (Strumenti) 6722
Epiche 6722
Risorse correlate 6725
Informazioni aggiuntive 6725
Allegati 6726
Altri modelli 6727
..... 6730

AWS Modelli di orientamento prescrittivi

I modelli di guida prescrittiva di Amazon Web Services (AWS) forniscono step-by-step istruzioni, architettura, strumenti e codice per implementare scenari specifici di migrazione, modernizzazione e distribuzione del cloud. Questi modelli, esaminati da esperti in materia AWS, sono pensati per i costruttori e gli utenti pratici che stanno pianificando o stanno per migrare verso. AWS Supportano anche gli utenti che sono già attivi AWS e stanno cercando modi per ottimizzare o modernizzare le proprie operazioni cloud.

Puoi utilizzare questi modelli per spostare i carichi di lavoro on-premise o cloud di varia complessità AWS e per accelerare gli sforzi di adozione, ottimizzazione e modernizzazione del cloud, indipendentemente dal fatto che tu sia nella fase di prototipazione, pianificazione o implementazione del progetto. Ad esempio, per un progetto di migrazione al cloud:

- Nella fase di pianificazione, puoi valutare le diverse opzioni disponibili verso cui migrare AWS. Puoi scegliere il modello più adatto alle tue esigenze, a seconda che tu voglia trasferire, riospitare, riorganizzare la piattaforma o riprogettare. Puoi anche comprendere i diversi strumenti disponibili per la migrazione e iniziare a pianificare l'acquisto di licenze o avviare conversazioni iniziali con i fornitori.
- Nelle fasi di dimostrazione del concetto e di implementazione, puoi seguire step-by-step le istruzioni fornite nello schema verso cui migrare il tuo carico di lavoro. AWS Ogni modello include dettagli come prerequisiti, architetture di riferimento di destinazione, strumenti, step-by-step attività, best practice, risoluzione dei problemi e codice.
- Se utilizzi già il Cloud AWS, puoi trovare modelli che ti aiuteranno a modernizzare, ottimizzare, scalare e proteggere l'uso delle risorse cloud.

Per visualizzare gli elenchi di pattern suddivisi per dominio tecnico, utilizza i seguenti link o le opzioni di filtro e ricerca nella home page di AWS Prescriptive Guidance.

- Fondamenti del cloud
- AI e apprendimento automatico
- Analisi
- Calcolo
- Database e archiviazione
- Strumenti per sviluppatori

- IoT
- Migrazione e modernizzazione
- Gestione
- Sicurezza, identità e conformità

Per visualizzare tutte le pubblicazioni, incluse guide, strategie e modelli, consulta la home page di AWS Prescriptive Guidance.

Fondamenti del cloud

Argomenti

- Automatizza la creazione di account utilizzando Landing Zone Accelerator su AWS
- Inventaria automaticamente AWS le risorse su più account e regioni
- Crea una AWS landing zone che includa MongoDB Atlas
- Configura i log di flusso VPC per la centralizzazione su Account AWS
- Gestisci i set di AWS autorizzazioni in modo dinamico utilizzando Terraform
- Etichetta automaticamente gli allegati Transit Gateway utilizzando AWS Organizations
- Altri modelli

Automatizza la creazione di account utilizzando Landing Zone Accelerator su AWS

Justin Kuskowski, Joe Behrens e Nathan Scott, Amazon Web Services

Riepilogo

Questo schema spiega come utilizzare Landing Zone Accelerator su AWS una soluzione per implementarne automaticamente una nuova quando un utente autorizzato invia una richiesta. Account AWS Viene utilizzato AWS Step Functions per orchestrare una serie di funzioni. AWS Lambda Le funzioni Lambda aggiungono le informazioni sull'account a un repository Git, avviano una AWS CodePipeline pipeline e convalidano il provisioning delle risorse necessarie. AWS Quando il processo è completo, l'utente riceve una notifica che l'account è stato creato.

Facoltativamente, puoi integrare gruppi di ID Microsoft Entra e assegnare set di AWS IAM Identity Center autorizzazioni durante il processo di creazione dell'account. Se l'organizzazione utilizza l'ID Microsoft Entra come fonte di identità, questa funzionalità opzionale consente di gestire e configurare automaticamente l'accesso al nuovo account.

Prerequisiti e limitazioni

Prerequisiti

- Accesso all'account di gestione in AWS Organizations
- AWS Cloud Development Kit (AWS CDK) versione 2.118.0 o successiva, installata e configurata
- Python versione 3.9 o successiva, installata
- AWS Command Line Interface (AWS CLI) versione 2.13.19 o successiva, installata
- Docker versione 24.0.6 o successiva, installata
- Landing Zone Accelerator on AWS solution, distribuito nell'account di gestione
- (Opzionale) Microsoft Entra ID e IAM Identity Center, integrati

Limitazioni

Il flusso di lavoro per la creazione dell'account supporta esecuzioni sequenziali per implementarne una singola. Account AWS Questa limitazione assicura che il flusso di lavoro per la creazione dell'account sia completato con successo senza dover competere per le risorse durante un'esecuzione parallela.

Architecture

Architettura Target

L'immagine seguente mostra l'architettura di alto livello che automatizza la creazione di un nuovo dispositivo Account AWS utilizzando Landing Zone Accelerator on. AWS AWS Step Functions orchestra l'automazione. Ogni attività nel flusso di lavoro Step Functions viene eseguita da una o più AWS Lambda funzioni.

Il diagramma mostra il flusso di lavoro seguente:

1. L'utente richiede un account eseguendo uno script Python o utilizzando Amazon API Gateway.
2. Il flusso di lavoro di Account Creation Orchestrator inizia in. AWS Step Functions
3. Il flusso di lavoro aggiorna il `account-config.yaml` file nell'archivio del codice sorgente. Inoltre avvia il Landing Zone Accelerator sulla AWS pipeline e controlla lo stato della pipeline. Questa pipeline crea e configura il nuovo account. Per ulteriori informazioni su come funziona, consulta la panoramica sull'architettura del Landing Zone Accelerator su. AWS
4. (Facoltativo) Quando la pipeline è completa, il flusso di lavoro verifica se il gruppo esiste in Microsoft Entra ID. Se il gruppo non esiste in Microsoft Entra ID, il flusso di lavoro aggiunge il gruppo a Microsoft Entra ID.
5. Il flusso di lavoro esegue passaggi aggiuntivi che la AWS soluzione Landing Zone Accelerator on non è in grado di eseguire. I passaggi predefiniti includono:
   - Creazione di un alias di account in AWS Identity and Access Management (IAM)
   - Allegare tag all'account in AWS Organizations
   - Creazione di AWS Systems Manager parametri in Parameter Store in base ai tag assegnati all'account
6. (Facoltativo) Il flusso di lavoro assegna uno o più set di autorizzazioni al gruppo di ID Microsoft Entra specificato in precedenza. I set di autorizzazioni consentono agli utenti del gruppo di accedere al nuovo account e consentono loro di eseguire le azioni configurate.
7. Una AWS Lambda funzione esegue test di controllo qualità e convalida. Convalida la creazione di risorse, verifica che i tag siano stati creati e verifica che le risorse di sicurezza siano state distribuite.
8. Il flusso di lavoro rilascia l'account e utilizza Amazon Simple Email Service (Amazon SES) per notificare all'utente che il processo è stato completato correttamente.

Per ulteriori informazioni sul flusso di lavoro Step Functions, vedere il diagramma del flusso di lavoro Step Functions nella sezione Informazioni aggiuntive di questo modello.

Applicazioni Microsoft Entra ID

Se si sceglie di effettuare l'integrazione con Microsoft Entra ID, si creano le due applicazioni seguenti quando si distribuisce questo modello:

- Un'applicazione collegata a IAM Identity Center e che assicura che i gruppi di ID Microsoft Entra siano disponibili in IAM Identity Center. In questo esempio, questa applicazione Microsoft Entra ID è denominata`LZA2`.
- Un'applicazione che consente a una funzione Lambda di comunicare con Microsoft Entra ID e richiamare Microsoft Graph. APIs In questo modello, questa applicazione è denominata`create_aws_account`.

Queste applicazioni raccolgono dati che vengono utilizzati per sincronizzare il gruppo di ID Microsoft Entra e assegnare set di autorizzazioni.

Tools (Strumenti)

Servizi AWS

- Amazon API Gateway ti aiuta a creare, pubblicare, gestire, monitorare e proteggere REST, HTTP e WebSocket APIs su qualsiasi scala. In questo modello, si utilizza API Gateway per verificare la disponibilità dei Account AWS nomi, avviare il AWS Step Functions flusso di lavoro e controllare lo stato dell'esecuzione di Step Functions.
- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che consente di definire e fornire Cloud AWS l'infrastruttura nel codice.
- AWS Control Towerti aiuta a configurare e gestire un ambiente AWS multi-account, seguendo le migliori pratiche prescrittive.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altro modo. Account AWS Questa soluzione utilizza una EventBridge regola che avvia una funzione Lambda se lo stato del flusso di lavoro Step Functions cambia in `FailedTimed-out`, o. `Aborted`
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.

- AWS IAM Identity Centerti aiuta a gestire centralmente l'accesso Single Sign-On (SSO) a tutte le tue applicazioni e a quelle sul cloud. Account AWS
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati. In questo modello, AWS KMS le chiavi vengono utilizzate per crittografare i dati, come i dati archiviati in Amazon Simple Storage Service (Amazon S3), le variabili di ambiente Lambda e i dati in Step Functions.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS Organizationsè un servizio di gestione degli account che ti aiuta a consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente.
- Amazon Simple Email Service (Amazon SES) Simple Email Service (Amazon SES) ti aiuta a inviare e ricevere e-mail utilizzando i tuoi indirizzi e-mail e domini. Dopo aver creato correttamente un nuovo account, ricevi una notifica tramite Amazon SES.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail. Se si verifica un errore durante il processo di creazione dell'account, Amazon SNS invia una notifica agli indirizzi e-mail che configuri.
- AWS Step Functionsè un servizio di orchestrazione senza server che ti aiuta a combinare AWS Lambda funzioni e altro Servizi AWS per creare applicazioni aziendali critiche.
- AWS Systems Manager Parameter Store fornisce uno storage sicuro e gerarchico per la gestione dei dati di configurazione e la gestione dei segreti.

Altri strumenti

- awscurl automatizza il processo di firma delle richieste AWS API e ti aiuta a effettuare richieste come comando curl standard.
- Microsoft Entra ID, precedentemente noto come Azure Active Directory, è un servizio di gestione delle identità e degli accessi basato sul cloud.
- Microsoft Graph ti APIs aiuta ad accedere ai dati e all'intelligence nei servizi cloud Microsoft, come Microsoft Entra e Microsoft 365.

Archivio di codice

Il codice per questo pattern è disponibile nel GitHub lza-account-creation-workflowrepository.

La directory lambda_layer contiene i seguenti livelli, a cui si fa riferimento in più di una funzione Lambda:

- account_creation_helper — Questo livello include i moduli per l'assunzione dei ruoli e il controllo dei progressi. AWS Service Catalog
- boto3 — Questo livello include il AWS SDK per Python (Boto3)modulo per assicurarsi che abbia la versione più recente. AWS Lambda
- identity_center_helper — Questo livello supporta le chiamate a IAM Identity Center.

La directory lambda_src contiene le seguenti funzioni Lambda:

- AccountTagToSsmParameter— Questa funzione utilizza i tag allegati all'account per creare parametri AWS Organizations in Parameter Store. Ogni parametro inizia con un `/account/tags/` prefisso.
- AttachPermissionSet— Questa funzione aggiunge un set di autorizzazioni a un gruppo IAM Identity Center.
- Azure ADGroup Sync: questa funzione sincronizza il gruppo ID Microsoft Entra di destinazione con IAM Identity Center.
- CheckForRunningProcesses— Questa funzione verifica se la `AWSAccelerator-Pipeline` pipeline è attualmente in esecuzione. Se la pipeline è in esecuzione, la funzione ritarda il flusso di lavoro. AWS Step Functions
- CreateAccount— Questa funzione utilizza AWS Service Catalog e crea AWS Control Tower il nuovo. Account AWS
- CreateAdditionalResources— Questa funzione crea AWS risorse che non sono gestite da Landing Zone Accelerator o AWS CloudFormation, ad esempio, l'alias e AWS Service Catalog i tag dell'account.
- GetAccountStatus— Questa funzione esegue la scansione del prodotto fornito per determinare se il processo AWS Service Catalog di creazione dell'account è stato completato.
- GetExecutionStatus— Questa funzione recupera lo stato di un'esecuzione in esecuzione o completata. AWS Step Functions
- NameAvailability— Questa funzione verifica se esiste già un Account AWS nome in AWS Organizations.
- ReturnResponse— Se la creazione dell'account è avvenuta con successo, questa funzione restituisce l'ID del nuovo account. Se la creazione dell'account non è andata a buon fine, viene restituito un messaggio di errore.

- RunStepFunction— Questa funzione esegue il AWS Step Functions flusso di lavoro che crea l'account.
- SendEmailWithSES — Questa funzione invia e-mail agli utenti che attendono il completamento della creazione dell'account.
- Convalida ADGroup SyncTo SSO: questa funzione verifica se i gruppi ID Microsoft Entra specificati sono sincronizzati con IAM Identity Center.
- ValidateResources— Questa funzione verifica che tutte le AWS Control Tower personalizzazioni siano state eseguite correttamente.

Best practice

Si consigliano le seguenti convenzioni di denominazione per: AWS CDK

- Avviate tutti i parametri con un prefissop.
- Inizia tutte le condizioni con un `c` prefisso.
- Avvia tutte le risorse con un `r` prefisso.
- Avvia tutte le uscite con un o prefisso.

Epiche

Implementa i ruoli IAM per la convalida e l'etichettatura

Operazione	Description	Competenze richieste
Prepara il Landing Zone Accelerator AWS per la personalizzazione.	1. Nel repository di codice di Landing Zone Accelerator on AWS code, crea un file denominato. `customizations-config.yaml` Questo file viene utilizzato per definire le personalizzazioni per la soluzione principale. Per ulteriori informazioni, vedere	AWS DevOps

Operazione	Description	Competenze richieste
	Personalizzazione della soluzione. 2. Nel `customizations-config.yaml` file, create una sezione chiamata `cloudFormationStacks` .	

Operazione	Description	Competenze richieste
Preparati a distribuire il `lza-account-creation-validation` ruolo.	Ora personalizzi la soluzione per implementare il ruolo `lza-account-creation-validation` IAM in tutti gli account diversi dall'account di gestione. Questo ruolo fornisce alla funzione `ValidateResources` Lambda l'accesso in sola lettura ai nuovi account. 1. Scarica il file.yaml da account-creation-validation-role. GitHub 2. Salvate il file nella posizione indicata nella sezione dei modelli del file. `customizations-config.yaml` 3. Apri il file `customizations-config.yaml` . 4. Nella `cloudFormationStacks` sezione, aggiungi il codice seguente. Aggiorna l'obiettivo Regione AWS secondo necessità per la tua landing zone: `- deploymentTargets: organizationalUnits: - Root excludedAccounts:`	AWS DevOps

Operazione	Description	Competenze richieste

	<pre>- Management
	
	descripti on: IAM Role to allow Account Validation
	name: lza-account-creation-validation
	regions:
	- us-east-1
	template: cloudformation/acc ount-creation-vali dation-role.yaml
	runOrder: 1
	terminati onProtection: true
	parameters:
	- name: pManagementAccount Id
	value: "{{ account Managemen t }}"</pre> 5. Salvare e chiudere il file `customizations-con fig.yaml` .	

Operazione	Description	Competenze richieste
Preparati a distribuire il `account-tagging-to-ssm-parameter-role` ruolo.	Ora personalizzi la soluzione per implementare il ruolo `account-tagging-to-ssm-parameter-role` IAM in tutti gli account diversi dall'account di gestione. Questo ruolo viene utilizzato per creare i parametri in AWS Systems Manager Parameter Store. 1. Scarica il file account-tagging-to-ssm-parameter-role.yaml da. GitHub 2. Salvate il file nella posizione indicata nella sezione dei modelli del file. `customizations-config.yaml` 3. Apri il file `customizations-config.yaml` . 4. Nella `cloudFormationStacks` sezione, aggiungi il codice seguente. Aggiorna l'obiettivo Regione AWS secondo necessità per la tua landing zone: `- deploymentTargets: organizationalUnits: - Root excludedAccounts: - Management`	AWS DevOps

Operazione	Description	Competenze richieste

	<pre> descripti
	on: IAM Role to
	 create SSM Parameter
	s based on Account
	 Tagging
	          name: lza-
	account-tagging-to-
	ssm-parameter
	          regions:
	            - us-
	east-1
	          template:
	 cloudformation/acc
	ount-tagging-to-ss
	m-parameter-role.y
	aml
	          runOrder: 1
	          terminati
	onProtection: true
	
	          parameters:
	            - name:
	 pManagementAccount
	Id
	              value:
	 "{{ account Managemen
	t }}"</pre> 5. Salvare e chiudere il file `customizations-config.yaml` .	

Operazione	Description	Competenze richieste
Preparati a distribuire il `config-log-validation-role` ruolo.	Ora personalizzi la soluzione per implementare il ruolo `config-log-validation-role` IAM nell'account di archiviazione dei log. Questo ruolo consente alla funzione `ValidateResources` Lambda di accedere al bucket Amazon S3 per le regole di registrazione e accesso. AWS Config 1. Scarica il file.yaml da. config-log-validation-role GitHub 2. Salvate il file nella posizione indicata nella sezione dei modelli del file. `customizations-config.yaml` 3. Apri il file `customizations-config.yaml` . 4. Nella `cloudFormationStacks` sezione, aggiungi il codice seguente. Aggiorna l'obiettivo Regione AWS secondo necessità per la tua landing zone: `- deploymentTargets: accounts: - LogArchive description: IAM Role to`	AWS DevOps

Operazione	Description	Competenze richieste
	validate Config and Logs name: lza-config-log-validation-role regions: - us-east-1 template: cloudformation/config-log-validation-role.yaml runOrder: 1 terminationProtection: true parameters: - name: pManagementAccountId value: "{{ account Management }}" 5. Salva, chiudi e conferma le modifiche apportate al `customizations-config.yaml` file.	

(Facoltativo) Recupera dati da Microsoft Entra ID

Operazione	Description	Competenze richieste
Crea l'applicazione che consente a una funzione Lambda di comunicare con Microsoft Entra ID.	1. Nell'interfaccia di amministrazione Microsoft Entra ID, registra l'`create_aws_account` applicazione. Per istruzioni, consulta	ID Microsoft Entra

Operazione	Description	Competenze richieste
	Registrare un'applicazione nella documentazione Microsoft. 2. Segui le istruzioni in Aggiornare le autorizzazioni richieste di un'app nella documentazione Microsoft per configurare le seguenti autorizzazioni Microsoft Graph per l'`create_aws_account` applicazione: • Applicazione. Leggi tutto • Tipo: Applicazione • È richiesto il consenso dell'amministratore: Sì • AppRoleAssignment. ReadWrite.Tutto • Tipo: Applicazione • È richiesto il consenso dell'amministratore: Sì • Gruppo. ReadWrite.Tutti • Tipo: Applicazione • È richiesto il consenso dell'amministratore: Sì • ServicePrincipalEndpoint.Leggi tutto • Tipo: Applicazione • È richiesto il consenso dell'amministratore: Sì • Sincronizzazione ReadWrite.Tutto • Tipo: Applicazione	

Operazione	Description	Competenze richieste

	• È richiesto il consenso dell'amministratore: Sì
	• User.Read
	  • Tipo: delegato
	  • È richiesto il consenso dell'amministratore: No	

Recupera i valori per l'`create_aws_account` applicazione.	Ora recuperate i valori necessari per l'`create_aws_account` applicazione.
	1. Nell'interfaccia di amministrazione di Microsoft Entra ID, vai a Registrazioni app, quindi scegli`create_aws_account` .
	2. Nel riquadro a sinistra, scegli Panoramica.
	3. Nella pagina Panoramica, prendi nota dei seguenti valori:
	  • ID dell'applicazione (client)
	  • ID della directory (tenant)
	4. Nel riquadro di sinistra, sotto Gestisci, scegli Certificati e segreti.
	5. Nella pagina Certificati e segreti, scegli la scheda Client secrets, quindi prendi nota dei seguenti valori:
	  • Valore segreto del client
	  • ID segreto del cliente
		ID Microsoft Entra

Operazione	Description	Competenze richieste
Crea l'applicazione che integra Microsoft Entra ID con IAM Identity Center.	Nell'interfaccia di amministrazione Microsoft Entra ID, registra l'`LZA2`applicazione. Per istruzioni, consulta Registrare un'applicazione nella documentazione Microsoft.	ID Microsoft Entra
Recupera i valori per l'`LZA2`applicazione.	Ora recuperate i valori necessari per l'`LZA2`applicazione. 1. Nell'interfaccia di amministrazione Microsoft Entra ID, accedi ad Applicazioni aziendali, quindi scegli`LZA2`. 2. Nel riquadro a sinistra, scegli Panoramica. 3. Nella pagina Panoramica, prendi nota dei seguenti valori: • Name • ID dell'oggetto 4. Nel riquadro di sinistra, in Gestisci, scegli Manifest. 5. Nel file JSON, nella `appRoles` sezione, individua il ruolo dell'app denominato`User`. 6. Prendi nota del `id` valore di questo ruolo dell'app.	ID Microsoft Entra

Operazione	Description	Competenze richieste

Crea un segreto.	1. Nel AWS CLI, inserisci il seguente comando per creare le variabili. Utilizzate i valori recuperati per le LZA2 applicazioni `create_aws_account` and:
	`# Variables for create_aws_account app TENANT_ID='<Directory ID>' CLIENT_ID='<Application ID>' SECRET_ID='<Client secret ID>' SECRET_VALUE='<Client secret value>'`
	`# Variables for LZA2 app OBJECT_ID='<Object ID>' APP_ROLE_ID='<App role ID>' ENTERPRISE_APP_NAME='<Name>'`
	2. Immettete il seguente comando per creare un segreto denominato `GraphApiSecret` in Gestione dei segreti AWS:
	`aws secretsmanager create-secret \ --name GraphApiSecret \`
		AWS DevOps

Operazione	Description	Competenze richieste

	`--secret-string "{\"client_id\": \"${CLIENT_ID}\", \"tenant_id\": \"${TENANT_ID}\", \"object_id\": \"${OBJECT_ID}\", \"app_role_id\": \"${APP_ROLE_ID}\", \"secret_value\": \"${SECRET_VALUE}\", \"secret_id\": \"${SECRET_ID}\"}"`
	
	Se è necessario aggiornare il segreto in futuro, è possibile aggiornare le variabili ed eseguire il seguente comando:
	
	`aws secretsmanager update-secret \ --secret-id GraphApiSecret \ --secret-string "{\"client_id\": \"${CLIENT_ID}\", \"tenant_id\": \"${TENANT_ID}\", \"object_id\": \"${OBJECT_ID}\", \"app_role_id\": \"${APP_ROLE_ID}\", \"secret_value\": \"${SECRET_VALUE}\", \"secret_id\": \"${SECRET_ID}\"}"`	

Implementazione della soluzione

Operazione	Description	Competenze richieste

Clona il codice sorgente.	1. Inserisci il seguente comando per clonare il lza-account-creation-workflowrepository:
	`git clone https://github.com/aws-samples/lza-account-creation-workflow`
	2. Immettete il seguente comando per navigare nella `config` directory del repository clonato:
	`cd lza-account-creation-workflow/config`
		DevOps ingegnere

Aggiorna il `deploy-config.yaml` file.	1. Apri il file `deploy-config.yaml` . 2. Rivedi il modello e aggiorna i valori secondo necessità per la distribuzione nel tuo AWS ambiente. Ad esempio, aggiorna i valori per quanto segue: • `accountCreationFailure` • `accountCompletionFromEmail` • `ssoLoginUrl` • `rootEmailPrefix` • `rootEmailDomain`	AWS DevOps

Operazione	Description	Competenze richieste
	3. Se stai effettuando l'integrazione con Microsoft Entra ID, procedi come segue: • Imposta `enableAzureADIntegration` su `true`. • Per `graphApiSecretName` valore, inserisci il segreto che hai creato in precedenza (`GraphApiSecret`). 4. Salvare e chiudere il file `deploy-config.yaml`.	

Operazione	Description	Competenze richieste

Implementa la soluzione nel tuo AWS ambiente.	1. Immetti il comando seguente:
	`cdk bootstrap <account-number>/<Region>`
	Per ulteriori informazioni, consulta Bootstrapping nella documentazione. AWS CDK
	2. Immettete il seguente comando per sintetizzare il modello: CloudFormation
	`cdk synth`
	Per ulteriori informazioni, consulta Configurare ed eseguire la sintesi AWS CDK dello stack nella documentazione. AWS CDK
	3. Immettere il comando seguente per distribuire la soluzione.
	`cdk deploy`
	Per ulteriori informazioni, consulta Distribuire AWS CDK applicazioni nella AWS CDK documenta zione.
		AWS DevOps

Operazione	Description	Competenze richieste
	Note Questa soluzione utilizza un bucket Amazon S3 per archiviare il codice sorgente di questa soluzione. Puoi utilizzare lo script upload_to_source_bucket.py per creare un archivio del codice sorgente e caricare una versione aggiornata.	

Opzione 1 — Crea un account usando Python

Operazione	Description	Competenze richieste
Identifica quali argomenti utilizzare.	Scegliete quali argomenti utilizzare quando eseguite lo script Python che avvia il flusso di lavoro Step Functions . Per un elenco completo degli argomenti, consultate la sezione Informazioni aggiuntive di questo modello.	AWS DevOps, Python
Avvia lo script Python.	1. Inserisci il seguente comando per navigare nel repository clonato:	DevOps ingegnere, Python

Operazione	Description	Competenze richieste

	`cd lza-account-creation-workflow`
	2. Esegui lo script Python che avvia il flusso di lavoro Step Functions. Di seguito è riportato un comando di esempio che include argomenti e valori di esempio:
	`python ./run-stepfunction.py \ --account-name "lza-test-01" \ --support-dl "johnsmith@example.com" \ --managed-org-unit "Workloads/Workload-1" \ --purpose "Testing new micro service" \ --force-update true \ --ad-integration "{\"CustomerAccountAdmin\": \"platform-admin\", \"CustomerAccountDev\": \"workload1-app1\"}" \ --bypass-creation true \ --tags APPLICATION=TestingMicroService`	

Operazione	Description	Competenze richieste
	3. Attendi di ricevere una notifica che indica che l'account è stato creato con successo. **Note** Se il processo di creazione dell'account non riesce, Amazon SNS invia una notifica agli indirizzi e-mail definiti `accountCreationFailure` nel `deploy-config.yaml` file. Il richiedente dell'account non viene avvisato.	

Opzione 2: creare un account utilizzando API Gateway e awscurl

Operazione	Description	Competenze richieste
Imposta le variabili per awscurl.	1. Inserisci il seguente comando per navigare nella directory del codice sorgente: `cd lza-account-creation-workflow` 2. Immettete il seguente comando per impostare	AWS DevOps

Operazione	Description	Competenze richieste

	le variabili della chiave di AWS accesso. È possibile copiare le variabili dal portale di AWS accesso e incollarle in una shell. Di seguito è riportato un esempio:
	`export AWS_ACCES S_KEY_ID="<id>" export AWS_SECRE T_ACCESS_KEY="<key >" export AWS_SESSI ON_TOKEN="<token>"`
	3. Immettete il seguente comando Regione AWS per impostare le chiamate API:
	`export AWS_REGION= $(aws configure get region)`
	4. Immettete il seguente comando per recuperar e l'endpoint API Gateway dall'`lza-account-creation-workflow-application` CloudForm ation output:
	`export API_GATEW AY_ENDPOINT=$(aws cloudformation describe-stacks -- stack-name "lza-acco unt-creation-workf`	

Operazione	Description	Competenze richieste
	low-application" --query 'Stacks[*].Outputs[?OutputKey==`oApiGatewayCreateAccountEndpoint`].OutputValue' --output text)	
Verifica la disponibilità dei nomi.	Immettere il comando seguente per verificare che il nome sia disponibile per Account AWS. Sostituisci <AWS_ACCOUNT_NAME> con il nome dell'account di destinazione: awscurl --service execute-api \ --region ${AWS_REGION} \ --access_key ${AWS_ACCESS_KEY_ID} \ --secret_key ${AWS_SECRET_ACCESS_KEY} \ --security_token ${AWS_SESSION_TOKEN} \ -X POST ${API_GATEWAY_ENDPOINT}check_name?account_name=<AWS_ACCOUNT_NAME>	AWS DevOps

Operazione	Description	Competenze richieste
Esegui il flusso di lavoro per la creazione dell'account.	1. Nella cartella principale del repository clonato, apri il file. `api_example.json` 2. Aggiorna i parametri con i tuoi valori di configurazione: `{ "account_name": "lza-test-01", "account_email": "johnsmith+lzatest01@example.com", "support_dl": "johnsmith@example.com", "managed_org_unit": "Workloads/Workload-1", "ad_integration": [{ "PermissionSetName": "CustomerAccountAdmin", "ActiveDirectoryGroupName": "platform-admin" }, { "PermissionSetName": "CustomerAccountDev", "ActiveDirectoryGroupName": "workload1-app1" }],`	AWS DevOps

Operazione	Description	Competenze richieste
	<pre>"force_update": "true", "bypass_creation": "false", "account_tags": [{ "Key": "Environment", "Value": "Dev" }, { "Key": "DeploymentMethod", "Value": "ApiGateway" }] }</pre> 3. Salvare e chiudere il file `api_example.json` . 4. Immettete il seguente comando per avviare il flusso di lavoro Step Functions: <pre>awscurl --service execute-api \ --data @api-example.json \ --region ${AWS_REGION} \ --access_key ${AWS_ACCESS_KEY_ID} \ --secret_key ${AWS_SECRET_ACCESS_KEY} \</pre>	

Operazione	Description	Competenze richieste

	<pre>--security_token
	${AWS_SESSION_TOKEN} \
	-X POST ${API_GATEWAY_ENDPOINT}execute</pre>
	5. Nell'output del comando precedente, prendi nota dell'esecuzione di Step Functions Amazon Resource Name (ARN).
	6. Se si desidera verificare lo stato del flusso di lavoro Step Functions, immettere il comando seguente. Sostituisci `<STEP_FUNCTION_EXECUTION_NAME>` con l'ARN o il nome di esecuzione di Step Functions:
	<pre>awscurl --service execute-api \
	--region ${AWS_REGION} \
	--access_key ${AWS_ACCESS_KEY_ID} \
	--secret_key ${AWS_SECRET_ACCESS_KEY} \
	--security_token ${AWS_SESSION_TOKEN} \
	-X GET ${API_GATEWAY_ENDPOINT}get_execution_status?execution=<STEP_FUN</pre>	

Operazione	Description	Competenze richieste
	`CTION_EXECUTION_NA ME>` 7. Attendi di ricevere una notifica che indica che l'account è stato creato con successo. **Note** Se il processo di creazione dell'acco unt non riesce, Amazon SNS invia una notifica agli indirizzi e-mail definiti `accountCr eationFailure` nel `deploy-co nfig.yaml` file. Il richiedente dell'account non viene avvisato.	

(Facoltativo) Pulisci la soluzione

Operazione	Description	Competenze richieste
Rimuovi oggetti dai bucket Amazon S3.	Rimuovi qualsiasi oggetto nei seguenti bucket Amazon S3: • `lza-account-creati on-work-<CDK_UNIQU E_ID>`	AWS DevOps

Operazione	Description	Competenze richieste
	• `lza-account-creation-workflow-src-<AWS_REGION>-<AWS_ACCOUNT>` • `lza-account-creation-workflow-<AWS_REGION>-<AWS_ACCOUNT>`	

Elimina lo CloudFormation stack.	Immettete il seguente comando per eliminare lo CloudFormation stack:
	`aws cloudformation delete-stack \ --stack-name lza-account-creation-workflow-application`
	`aws cloudformation wait stack-delete-complete \ --stack-name lza-account-creation-workflow-application`
		AWS DevOps

Elimina la pipeline	Inserisci il seguente comando per eliminare la `lza-account-creation-workflow-pipeline` pipeline: `cdk destroy lza-account-creation-workflow-pipeline --force`	AWS DevOps

Risorse correlate

- Landing Zone Accelerator su AWS (libreria di AWS soluzioni)
- Risoluzione dei AWS CDK problemi più comuni (AWS CDK documentazione)

Informazioni aggiuntive

Diagramma del flusso di lavoro Step Functions

L'immagine seguente mostra gli stati nel flusso di lavoro Step Functions.

Arguments (Argomenti)

Di seguito sono riportati gli argomenti che è possibile utilizzare quando si esegue lo script Python che avvia il flusso di lavoro Step Functions.

I seguenti argomenti sono obbligatori:

- `account-name (-a)`(stringa) — Il nome del nuovo. Account AWS
- `support-dl (-s)`(stringa) — L'indirizzo e-mail che riceve una notifica quando il processo di creazione dell'account è completo.
- `managed-org-unit (-m)`(stringa) — L'unità organizzativa gestita (OU) che conterrà il nuovo account.

I seguenti argomenti sono facoltativi:

- `ad-integration (-ad)`(dizionario di stringhe) — Il gruppo di ID Microsoft Entra e il set di autorizzazioni assegnato. Di seguito è riportato un esempio di utilizzo di questo argomento:

```
--ad-integration "{\"<PermissionSetName>\": \"<EntraIdGroupName>\"}"
```

- `account-email (-e)`(stringa) — L'indirizzo e-mail dell'utente root del nuovo Account AWS.

> **Note**
>
> Se questo argomento non viene utilizzato, verrà generato un indirizzo e-mail utilizzando i valori `rootEmailPrefix` e `rootEmailDomain` dal `configs/deploy-config.yaml`

file. Se non viene fornito un indirizzo e-mail, viene generato un indirizzo e-mail utilizzando il seguente formato:`rootEmailPrefix+accountName@rootEmailDomain`.

- `region (-r)`(stringa) — Il Regione AWS luogo in cui è stato distribuito il flusso di lavoro Step Functions. Il valore predefinito è `us-east-1`.
- `force-update (-f)`(string Boolean) — Immettere `true` per AWS Service Catalog forzare l'aggiornamento del prodotto fornito.
- `bypass-creation (-b)`(string Boolean) — Immettere `true` per ignorare l'aggiunta dell'account al `accounts-config.yaml` file e ignorare l'esecuzione della pipeline. `AWSAccelerator-Pipeline` Questo argomento viene in genere utilizzato per testare il processo di creazione dell'account o per eseguire il resto dei passaggi di Step Functions se si verifica un errore nella `Landing Zone Accelerator` pipeline.
- `tags (-t)`(stringa) — Tag aggiuntivi che si desidera aggiungere a. Account AWS Per impostazione predefinita, vengono aggiunti i seguenti tag: `account-namesupport-dl`, `epurpose`. Di seguito è riportato un esempio di utilizzo di questo argomento:

```
--tags TEST1=VALUE1 TEST2=VALUE2
```

Inventaria automaticamente AWS le risorse su più account e regioni

Matej Macek, Amazon Web Services

Riepilogo

Questo modello delinea un approccio automatizzato per mantenere un inventario completo di AWS risorse su più account e. Regioni AWSÈ progettato per aiutare gli ingegneri dell'infrastruttura e della sicurezza a migliorare le loro pratiche di gestione delle risorse. Viene utilizzato AWS Config per tenere traccia delle modifiche alle risorse, Amazon Athena per le interrogazioni e Amazon Quick Sight per dashboard interattivi. Implementa questa soluzione distribuendo uno stack. AWS CloudFormation

Questa soluzione è simile a quella presentata in Visualizzazione dei AWS Config dati con Amazon Athena e Amazon Quick AWS Sight (post di blog). Questo modello amplia tale soluzione per soddisfare i seguenti requisiti comuni e fornire i seguenti vantaggi chiave:

- Incentrato sulla conformità : questo approccio può aiutarvi a soddisfare requisiti normativi come PCI DSS, NIST SP 800-53, ISO/IEC 27001, HIPAA, GDPR e altri che impongono inventari accurati degli asset.
- Framework di personalizzazione: fornisce una base per la creazione di dashboard Quick Sight per varie risorse, in modo da poter personalizzare la soluzione in base alle proprie esigenze specifiche. AWS
- Miglioramenti basati sull'utente: questo approccio incorpora il feedback proveniente da casi d'uso reali e risponde alle richieste di una soluzione più completa.

I team di infrastruttura, sicurezza e finanza spesso affrontano problemi di visibilità e collaborazione in ambienti dinamici, con più account o più regioni. Questa soluzione è progettata per affrontare queste sfide e ridurre in modo significativo il tempo e gli sforzi necessari per creare e mantenere un inventario delle risorse. Il risultato è una visione centralizzata delle risorse che consente di migliorare le decisioni di allocazione delle risorse, identificare e mitigare i rischi, ottimizzare i costi e migliorare la visibilità e la collaborazione complessive. Questo approccio colma il divario tra le soluzioni concettuali e le esigenze di implementazione nel mondo reale per scopi operativi, di sicurezza e di conformità.

Prerequisiti e limitazioni

Prerequisiti

- Sono attivi i seguenti: Account AWS
  - Account di gestione: un account centralizzato per la fatturazione, la creazione di account e il controllo degli accessi all'interno dell'organizzazione
  - Account di controllo: un hub centralizzato per il monitoraggio della sicurezza, i controlli di conformità e le notifiche di deviazione
  - Account di archiviazione dei log: un account centralizzato per l'archiviazione e l'analisi dei dati raccolti
- Nell'account di controllo, un AWS Config aggregatore che raccoglie e aggrega i dati di configurazione dagli account e dalle regioni di destinazione
- Nell'account di archiviazione dei log, imposta quanto segue:
  - Un bucket Amazon Simple Storage Service (Amazon S3) in cui archiviare i dati dall'aggregatore AWS Config
  - Un abbonamento Amazon Quick
  - Una connessione autorizzata tra Quick Sight e Amazon Athena
  - Autorizzazioni per accedere al bucket Amazon S3 tramite una query Athena
- AWS Command Line Interface (AWS CLI), installato e configurato
- Autorizzazioni per distribuire uno CloudFormation stack che fornisce le seguenti risorse:
  - Una funzione AWS Lambda
  - Una configurazione di notifica di Amazon S3
  - Database, tabelle e viste Athena
  - Set di dati e fonti di dati Quick Sight
- Autorizzazioni per eseguire automazioni in AWS Systems Manager
- Autorizzazioni per accedere a Quick

Limitazioni

- La soluzione si basa su. AWS Config AWS Config in genere registra le modifiche alla configurazione delle risorse subito dopo che viene rilevata una modifica o alla frequenza specificata. Tuttavia, questa operazione viene effettuata con la massima diligenza possibile e a volte può richiedere più tempo.
- Questa soluzione tiene traccia solo dei tipi di risorse AWS Config supportati.
- La soluzione non tiene traccia dell'inventario delle risorse di altri provider cloud o ambienti locali.

- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per informazioni sulla disponibilità regionale, consulta la pagina Endpoint e quote del servizio nella AWS documentazione e scegli il link relativo al servizio.

Architecture

Il diagramma seguente mostra un processo semplificato per la raccolta, l'organizzazione, l'analisi e la visualizzazione dei dati di configurazione e conformità tra più account di un'organizzazione. AWS

Il diagramma mostra il flusso di lavoro seguente:

1. In base a una pianificazione periodica, l' AWS Config aggregatore raccoglie i dati di configurazione e conformità relativi alle risorse negli account e nelle regioni di destinazione, quindi invia i dati al bucket Amazon S3 nell'account di archiviazione dei log.
2. L'aggiunta di nuovi AWS Config dati al bucket Amazon S3 richiama una funzione. AWS Lambda
3. La funzione Lambda partiziona i dati configurando chiavi con valori che corrispondono alla regione e alla data di ogni file di istantanea. Ciò consente di interrogare AWS Glue ed elaborare in modo efficiente i dati di configurazione e conformità.
4. Amazon Athena utilizza uno AWS Glue schema per eseguire query SQL sui dati archiviati nel bucket Amazon S3. Utilizza i metadati dello schema AWS Glue per comprendere la struttura dei dati.
5. Le viste in Athena definiscono ed estraggono i set di dati di destinazione.
6. Le dashboard di Quick Sight ti aiutano a visualizzare e analizzare i set di dati.

Tools (Strumenti)

Servizi AWS

- Amazon Athena è un servizio di query interattivo che ti aiuta ad analizzare i dati direttamente in Amazon S3 utilizzando SQL standard.
- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita in e. Account AWS Regioni AWS
- AWS Configfornisce una visione dettagliata delle risorse presenti Account AWS e di come sono configurate. Ti aiuta a identificare in che modo le risorse sono correlate tra loro e in che modo le

loro configurazioni sono cambiate nel tempo. Un AWS Config aggregatore raccoglie dati di AWS Config configurazione e conformità da più Account AWS regioni.

- AWS Glueè un servizio di estrazione, trasformazione e caricamento (ETL) completamente gestito. Ti aiuta a classificare, pulire, arricchire e spostare i dati in modo affidabile tra archivi di dati e flussi di dati. Questo modello utilizza un catalogo AWS Glue dati e un registro degli schemi.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS Organizationsè un servizio di gestione degli account che ti aiuta a consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente.
- Amazon Quick Sight è un servizio di business intelligence (BI) che ti aiuta a trasformare i dati grezzi in informazioni significative attraverso visualizzazioni interattive, dashboard e report. Quick Sight è un componente fondamentale di Amazon Quick.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Systems Managerti aiuta a gestire le applicazioni e l'infrastruttura in esecuzione in Cloud AWS. Semplifica la gestione delle applicazioni e delle risorse, riduce i tempi di rilevamento e risoluzione dei problemi operativi e aiuta a gestire le AWS risorse in modo sicuro su larga scala. AWS Systems Manager L'automazione semplifica le attività comuni di manutenzione, implementazione e riparazione per molti. Servizi AWS

Archivio di codice

Il AWS CloudFormation modello per questo pattern è disponibile nel repository di AWS Config visualizzazione GitHub . Questo CloudFormation modello distribuisce un runbook di AWS Systems Manager automazione configurato AWS Config per l'uso con Amazon Athena. Questa automazione si prepara AWS Glue a connettersi con il bucket Amazon S3 designato, crea viste in Amazon Athena e configura Quick Sight per la visualizzazione del dashboard.

Best practice

- Ti consigliamo di seguire le best practice illustrate nella sezione Configurazione e gestione di un ambiente sicuro con più account con on Prescriptive Guidance. AWS AWS Control Tower AWS
- Ti consigliamo di creare un AWS Config aggregatore che raccolga i dati di configurazione e conformità per l'intera organizzazione. AWS Per ulteriori informazioni, consulta Aggregazione di dati multiaccount e più regioni nella documentazione. AWS Config

- Prima di distribuire questa soluzione, ti consigliamo di consultare le informazioni sui prezzi correnti per Amazon S3 AWS Config, Athena e Quick.

Epiche

Distribuisci lo stack CloudFormation

Operazione	Description	Competenze richieste
Scarica il CloudFormation modello.	Scarica il modello Config-QuickSight CloudFormation - Visualization-SSM-Automation.yaml.	Amministratore AWS, amministratore cloud, DevOps ingegnere
Modifica il CloudFormation modello.	Completa questo passaggio solo se utilizzi AWS Control Towered AWS Config è gestito da AWS Control Tower. Devi modificare il CloudFormation modello. 1. Accedi all'account di gestione . 2. Apri la AWS Organizations console. 3. Vai alla pagina Settings (Impostazioni). Questa pagina mostra i dettagli sull'organizzazione, incluso l'ID dell'organizzazione. 4. Copia l'ID dell'organizzazione. 5. Nel vostro editor di testo preferito, aprite il file Config-QuickSight -Visualization-SSM-Automation.yaml.	DevOps ingegnere, amministratore AWS

Operazione	Description	Competenze richieste

	6. Trova la riga seguente:
	
	`return re.match( '^AWSLogs/(\d+)/Config/([\w-]+)/(\d+)/(\d+)/(\d+)/(\d+)/ConfigSnapshot/[^\\\]+$', object_key)`
	
	7. Sostituisci questa riga con la seguente, <ORGANIZATION_ID> dov'è l'ID che hai precedentemente copiato:
	
	`return re.match( '^<ORGANIZATION_ID>/AWSLogs/(\d+)/Config/([\w-]+)/(\d+)/(\d+)/(\d+)/(\d+)/ConfigSnapshot/[^\\\]+$', object_key)`
	
	8. Salvate e chiudete il file Config- QuickSight - Visualization-SSM-Automation.yaml.	

Operazione	Description	Competenze richieste
Crea uno CloudFormation stack.	Segui le istruzioni riportate in Creare uno stack dalla CloudFormation console. Tenere presente quanto segue: 1. Scegli Carica un file modello, quindi scegli il file YAML che hai scaricato. 2. In Nome stack, immetti `Config-QuickSight-Visualization-SSM-Automation` . 3. Seleziona Invia.	Amministratore AWS, amministratore cloud, DevOps ingegnere

Esegui l'automazione in Systems Manager

Operazione	Description	Competenze richieste
Trova il tuo nome utente Quick.	1. Apri la console Quick. 2. Apri il menu del profilo. 3. Prendi nota del nome utente. Questo valore ti servirà più tardi.	Amministratore AWS, amministratore cloud, DevOps ingegnere
Trova il nome del canale di distribuzione e il nome del bucket Amazon S3.	1. Nel AWS CLI, inserisci il seguente comando: `aws configservice describe-delivery-channels` 2. Prendi nota del nome del bucket Amazon S3 e del tuo canale di distribuzione.AWS	Amministratore AWS, amministratore cloud, DevOps ingegnere

Operazione	Description	Competenze richieste
	Config Questi valori ti serviranno in seguito.	

Operazione	Description	Competenze richieste
Esegui l'automazione in Systems Manager.	1. Apri la AWS Systems Manager console. 2. Nel pannello di navigazione, scegli Documenti. 3. Scegliere Owned by me (Di mia proprietà). 4. Scegliete Config- -Visualization. QuickSight 5. Scegli Esegui automazione. 6. Nella sezione Parametri di input, inserite i valori per i seguenti parametri: • `ConfigDeliveryChannelName` — Inserisci il nome del tuo canale AWS Config di distribuzione. Questo parametro è obbligatorio. • `ConfigS3BucketLocation` — Inserisci il nome del bucket Amazon S3 in cui archiviare AWS Config i dati di configurazione. Questo parametro è obbligatorio. • `QuickSightUserName` — Inserisci un nome utente con accesso amministrativo a Quick. Questo parametro è obbligatorio. • `AutomationAssumeRole` — L'Amazon	Amministratore AWS, amministratore cloud, DevOps ingegnere

Operazione	Description	Competenze richieste
	Resource Name (ARN) del ruolo AWS Identity and Access Managemen t (IAM) che consente a Systems Manager Automation di eseguire le azioni per tuo conto. Questo parametro è facoltativo. Lascia vuoto questo parametro. • `DeleteConfigVisual ization` — Sceglifalse. 7. Scegli Esegui.	

Visualizza i dati in Quick Sight

Operazione	Description	Competenze richieste
Aggiorna i dati.	Per pianificare gli aggiornam enti dei set di dati in base ai requisiti specifici, segui le istruzioni in Aggiornamento dei dati SPICE.	Amministratore AWS, DevOps ingegnere, amministratore cloud
Creare un'analisi in .	Per creare una dashboard in Quick Sight che ti aiuti a visualizzare le risorse, segui le istruzioni in Avvio di un'analisi in Quick Sight.	Amministratore di Quick Suite
Crea una dashboard.	1. Dopo aver modificato l'analisi Quick Sight, segui le istruzioni in Publishing	Amministratore di Quick Suite

Operazione	Description	Competenze richieste
	dashboard per creare una dashboard. Una dashboard è un'analisi che puoi condividere con altri utenti di Quick. 2. Segui le istruzioni riportate in Concedere l'accesso a una dashboard per condividere la dashboard con gli utenti Quick target.	

(Facoltativo) Pulizia

Operazione	Description	Competenze richieste
Eliminare le risorse create dall'automazione Systems Manager.	1. Apri la AWS Systems Manager console. 2. Nel pannello di navigazione, scegli Documenti. 3. Scegliere Owned by me (Di mia proprietà). 4. Scegliete Config- -Visualization. QuickSight 5. Scegli Esegui automazione. 6. Nella sezione Parametri di input, per il `DeleteConfigVisualization` parametro, immettete. `true` 7. Scegli Esegui.	Amministratore AWS, amministratore cloud, DevOps ingegnere
Elimina lo CloudFormation stack.	Per eliminare le risorse nello `Config-QuickSight-Visualization-SSM-`	Amministratore AWS, amministratore cloud, DevOps ingegnere

Operazione	Description	Competenze richieste
	`Automation` stack, segui le istruzioni in Eliminare uno stack dalla console. CloudFormation	

Risoluzione dei problemi

Problema	Soluzione

Amazon Quick sta tentando di connettersi a `us-east-1` Regione AWS, ma la creazione di risorse in quella regione non è consentita.	Una politica di controllo del servizio limita il tuo abbonamento ad Amazon Quick in questa regione. Nella politica di controllo del servizio, specifica manualmente l'obiettivo Regione AWS. Sostituisci `<REGION_ID>` con l'identif icatore di regione appropriato:
	`https://<REGION_ID>.quicksight.aws.amazon.com/sn/start/dashboards`
	Di seguito è riportato un esempio:
	`https://eu-central-1.quicksight.aws.amazon.com/sn/start/dashboards`

In Amazon Athena, viene visualizzato il seguente messaggio: `Before you run your first query, you need to set up a query result location in Amazon S3.`	Assicurati di aver preparato un bucket Amazon S3 in cui archiviare i risultati delle query di Amazon Athena. Segui quindi le istruzioni in Specificare una posizione per i risultati della query utilizzando la console Amazon Athena.

Risorse correlate

AWS documentazione

- AWS Config documentazione
- Documentazione Amazon Quick

AWS post sul blog

- Automatizza la visualizzazione AWS Config dei dati con AWS Systems Manager
- Come registrare periodicamente le modifiche alla configurazione delle risorse con AWS Config

Altre risorse

- Centro di apprendimento della community Amazon Quick
- Galleria della community di Amazon Quick

Crea una AWS landing zone che includa MongoDB Atlas

Igor Alekseev, Amazon Web Services

Anuj Panchal, MongoDB

Riepilogo

Questo modello descrive come creare una AWS landing zone integrata con un cluster MongoDB Atlas. L'infrastruttura viene implementata automaticamente utilizzando uno script Terraform.

Un AWS ambiente ben strutturato e multi-account, chiamato landing zone, offre scalabilità e sicurezza, in particolare per le aziende. Serve come base per una rapida implementazione di carichi di lavoro e applicazioni e contribuisce a garantire la fiducia nella sicurezza e nell'infrastruttura. La costruzione di una landing zone richiede un'attenta considerazione dei fattori tecnici e commerciali, tra cui la struttura degli account, il networking, la sicurezza e la gestione degli accessi. Queste considerazioni devono essere allineate agli obiettivi aziendali e di crescita futuri dell'organizzazione.

I casi d'uso di questo modello includono quanto segue.

- Piattaforme SaaS e PaaS aziendali: le applicazioni software as a service (SaaS) multitenant e le piattaforme platform as a service (PaaS) che funzionano su AWS possono utilizzare questa configurazione per fornire un accesso privato e sicuro a MongoDB Atlas senza esporre i dati sulla rete Internet pubblica.
- Settori altamente regolamentati: i carichi di lavoro bancari, finanziari, sanitari e governativi che richiedono una stretta conformità a standard come Health Insurance Portability and Accountability Act (HIPAA), Payment Card Industry Data Security Standard (PCI DSS), System and Organization Controls 2 () e General Data Protection Regulation (GDPRSOC2) beneficiano di:
  - Connettività privata crittografata tramite AWS PrivateLink
  - Alta disponibilità Multi-AZ di set di repliche MongoDB
- AI/ML Carichi di lavoro sicuri: le pipeline di formazione o inferenza in Amazon Bedrock, SageMaker Amazon AI o modelli di intelligenza artificiale personalizzati possono recuperare e archiviare dati in modo sicuro in MongoDB Atlas over. PrivateLink
- Disaster recovery e continuità aziendale: il design Multi-AZ garantisce che nessun guasto in una singola zona di disponibilità interferisca con i carichi di lavoro. Una replica Atlas impostata su tutte le zone di disponibilità garantisce il failover automatico. Questo è fondamentale per servizi sempre attivi come le app di tecnologia finanziaria (fintech), il digital banking o il monitoraggio sanitario.

Prerequisiti e limitazioni

Prerequisiti

- Accesso del proprietario dell'organizzazione a MongoDB Atlas in modo da poter creare chiavi API Atlas. Per informazioni su questo requisito, consulta Gestire l'accesso all'organizzazione nella documentazione di MongoDB.
- Un attivo. Account AWS
- Terraform, installato e configurato.
- Un cluster MongoDB Atlas, creato con MongoDB versione 6.0 o successiva.
- Familiarità con MongoDB e MongoDB Atlas. Per ulteriori informazioni, consulta la documentazione di MongoDB Atlas.

Limitazioni

- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Architecture

Il seguente diagramma dell'architettura di riferimento illustra la configurazione dell'implementazione per una AWS landing zone integrata con un endpoint privato MongoDB Atlas. Questa architettura di riferimento dimostra come stabilire una AWS landing zone sicura, scalabile e ad alta disponibilità integrata con MongoDB Atlas. Combinando le AWS migliori pratiche come l'implementazione Multi-AZ, i controlli di sicurezza con privilegi minimi e la connettività privata, questo design consente alle organizzazioni di fornire un ambiente robusto per le applicazioni moderne.

Questa architettura è composta da quanto segue:

VPC

- Un singolo cloud privato virtuale (VPC) si estende su tre zone di disponibilità.
- Il VPC è suddiviso in sottoreti allineate a ciascuna zona di disponibilità. Queste sottoreti distribuiscono i carichi di lavoro per garantire un'elevata disponibilità.

Accesso a Internet

- Un gateway Internet fornisce connettività Internet in uscita per le risorse che ne hanno bisogno, come applicazioni o host bastion.
- Le sottoreti pubbliche possono ospitare gateway NAT, che consentono ai carichi di lavoro di sottoreti private di scaricare aggiornamenti, patch e altri pacchetti necessari senza esporli direttamente alla rete Internet pubblica.

Sottoreti private e tabelle di routing

- I componenti delle applicazioni, i microservizi o altre risorse sensibili risiedono in genere in sottoreti private.
- Le tabelle di routing dedicate controllano i flussi di traffico. Instrada il traffico in uscita dalle sottoreti private ai gateway NAT per un accesso Internet sicuro e solo in uscita.
- Le richieste in entrata da Internet fluiscono attraverso sistemi di bilanciamento del carico elastici o host bastion (se utilizzati) nelle sottoreti pubbliche, quindi vengono indirizzate in modo appropriato verso le risorse di sottorete private.

Connettività MongoDB Atlas tramite PrivateLink

- L'architettura utilizza PrivateLink (tramite un endpoint VPC) la connessione sicura a MongoDB Atlas senza esporre i dati alla rete Internet pubblica.
- Le richieste rimangono sulla rete dorsale. AWS I dati in transito traggono vantaggio dalla PrivateLink crittografia e non vengono mai instradati sulla rete Internet pubblica.
- Il VPC dedicato MongoDB Atlas ospita i nodi primari e secondari e fornisce un ambiente sicuro e isolato per il cluster di database gestito.

Multi-AZ deployment (Implementazione Multi-AZ)

- I componenti critici dell'infrastruttura (come i gateway NAT e le sottoreti delle applicazioni) sono distribuiti su almeno tre zone di disponibilità. Se in una zona di disponibilità si verifica un'interruzione, questa architettura garantisce che i carichi di lavoro nelle restanti zone di disponibilità rimangano operativi.
- MongoDB Atlas, per impostazione predefinita, offre un'elevata disponibilità tramite set di repliche e garantisce che il livello di database rimanga tollerante ai guasti. L'infrastruttura critica è distribuita su almeno tre zone di disponibilità per garantire la resilienza.

Tools (Strumenti)

Servizi AWS

- Gestione dei segreti AWSti aiuta a sostituire le credenziali codificate nel codice, comprese le password, con una chiamata API per recuperare il segreto a livello di codice.

Altri prodotti e strumenti

- MongoDB Atlas è un database as a service (DBaaS) completamente gestito per l'implementazione e la gestione di database MongoDB nel cloud.
- Terraform è uno strumento di infrastruttura come codice (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali. In questo modello, si utilizza Terraform per eseguire uno script per facilitare la distribuzione delle risorse richieste su MongoDB AWS Atlas.

Archivio di codice

Il codice per questo pattern è disponibile nel repository MongoDB Atlas Landing Zone.AWS GitHub

Epiche

Scoperta e valutazione complete

Operazione	Description	Competenze richieste
Identifica le principali parti interessate.	Identifica tutte le principali parti interessate e i membri del team coinvolti nel tuo progetto di landing zone. Ciò potrebbe includere ruoli come: • Amministratori di database () DBAs • DevOps ingegneri • sviluppatori di applicazioni • Architetti di applicazioni	Responsabile della migrazione

Operazione	Description	Competenze richieste
Crea un progetto strutturale.	Crea un modello che delinei la struttura desiderata della tua landing zone e di quella abilitata per AWS MongoDB Atlas.	Responsabile della migrazione
Crea un piano di architettura.	Collabora con gli architetti delle applicazioni per analizzar e i requisiti e progettare un'architettura resiliente e tollerante ai guasti. Questo modello fornisce un modello di architettura iniziale come riferimento. È possibile personalizzare questo modello per soddisfare le esigenze di sicurezza e infrastruttura dell'organizzazione.	Architetto del cloud
Pianifica la configurazione e l'implementazione.	Determina, con tutte le parti interessate, come verrà implementata l'architettura, come verranno implement ate le misure di sicurezza e qualsiasi altro aspetto per garantire l'allineamento con gli interessi dell'organizzazione e del team richiedente.	Responsabile della migrazion e, ingegnere, DBA DevOps

Configurare l'ambiente MongoDB Atlas

Operazione	Description	Competenze richieste
Clonare il repository.	Clona il codice dal GitHub repository eseguendo il comando: `git clone https://github.com/mongodb-partners/AWS-MongoDB-Atlas-Landing-Zone`	Sviluppatore di app, ingegnere DevOps
Ottieni l'ID della tua organizzazione Atlas.	1. Se non disponi di un account MongoDB Atlas, registrane uno. 2. Segui i passaggi nella documentazione di MongoDB per creare un'organizzazione. 3. Copia l'ID dell'organizzazione.	DBA
Genera chiavi API Atlas a livello di organizzazione.	Per generare le tue chiavi API a livello di organizzazione in Atlas, segui le istruzioni nella documentazione di MongoDB.	DBA
Crea un segreto in Gestione dei segreti AWS.	Archivia le chiavi API MongoDB Atlas generate nel passaggio precedente come segreto chiave-valore in Secrets Manager. Per istruzioni, consulta la documentazione di Secrets Manager.	DevOps ingegnere

Operazione	Description	Competenze richieste
Seleziona il livello del cluster Atlas.	Per selezionare il livello di cluster Atlas corretto, segui le istruzioni nella documenta zione di MongoDB.	DBA

Configura gli ambienti AWS

Operazione	Description	Competenze richieste
Modifica lo script Terraform.	Nella copia locale del GitHub repository, aggiorna il nome segreto nel modules/m ongodb-atlas/mainfile.tf (riga 12), in modo che Terraform possa recuperare le credenzia li da Secrets Manager durante la distribuzione.	DevOps ingegnere
Crea un ID chiave di AWS accesso e una chiave segreta.	Per creare l'ID della chiave di AWS accesso e la chiave segreta, segui le istruzioni contenute nell'articolo AWS Re:post Come posso creare una chiave di AWS accesso? È consigliabile assegnare le politiche con i privilegi minimi necessari, ma in questo caso, selezionare la politica. `AdministratorAccess` Dopo aver creato la chiave di accesso, consulta le migliori pratiche di sicurezza in IAM per conoscere le migliori	DevOps ingegnere

Operazione	Description	Competenze richieste
	pratiche per la gestione delle chiavi di accesso.	
Alloca indirizzi IP elastici.	Alloca almeno due indirizzi IP elastici. IDs Per istruzioni, consulta la documentazione di Amazon Virtual Private Cloud (Amazon VPC).	DevOps ingegnere
Crea un bucket S3.	Crea un bucket S3 per archiviare lo stato della tua implementazione Terraform seguendo le istruzioni nella documentazione di Amazon Simple Storage Service (Amazon S3).	DevOps ingegnere

Operazione	Description	Competenze richieste
Aggiorna il bucket S3 per l'archiviazione.	Aggiorna le informazioni sul bucket S3 nella tua versione locale di environments/development/main.tf in modo che corrispondano al nome e alla regione del bucket che hai creato nel passaggio precedente e specifica un key prefix. Esempio: `terraform { ... backend "s3" { bucket = "startup-name-product-terraform" key = "network/dev" region = "ap-southeast-1" } }` Per questo esempio, puoi configurare Terraform per utilizzare il key `network/dev` prefix per organizzare il file di stato Terraform. È possibile modificare il valore in `prod` o in modo che `staging` corrisponda all'ambiente che si desidera creare. Per informazioni sull'utilizzo di più ambienti, consulta l'ultimo passaggio di questa sezione.	DevOps ingegnere

Operazione	Description	Competenze richieste
	Per ulteriori informazioni sui prefissi chiave di Amazon S3, consulta Organizzazione degli oggetti utilizzando i prefissi nella documentazione di Amazon S3.	
Imposta le variabili Terraform.	La landing zone di esempio definisce i valori delle variabili di input utilizzando i file di definizione variabile Terraform . Il file variabile si trova in environments/development/variables.tf. È possibile impostare i valori delle variabili nel environments/development/terraformfile.tfvars. Configura queste variabili come descritto nel file Readme per il repository. GitHub	DevOps ingegnere

Operazione	Description	Competenze richieste
Imposta le variabili di ambiente.	Se hai intenzione di eseguire lo script Terraform sul tuo computer locale, imposta le seguenti variabili di ambiente: • `AWS_ACCESS_KEY_ID` : ID della chiave di AWS accesso • `AWS_SECRET_ACCESS_KEY` : chiave di accesso AWS segreta • `AWS_DEFAULT_REGION` : Regione AWS • `TF_LOG`: livello di registro Terraform (`DEBUGoINFO`) Per ulteriori informazioni sull'impostazione delle variabili di ambiente, consulta la documentazione AWS Command Line Interface (AWS CLI).	DevOps ingegnere
Controlla le configurazioni del VPC.	Per seguire le migliori pratiche consigliate da AWS, configura le impostazioni per VPC e sottorete CIDRs, gateway NAT, percorsi e tabelle di routing nello script Terraform per soddisfare le esigenze della tua organizzazione. Per informazioni specifiche, consulta il file Readme per il repository. GitHub	DevOps ingegnere

Operazione	Description	Competenze richieste
Aggiunta di tag alle risorse .	Puoi taggare AWS le tue risorse per monitorarle quando vengono distribuite dallo script Terraform. Per esempi, consulta il file Readme per il repository. GitHub Per informazioni sul monitorag gio delle risorse tramite tag per costi, utilizzo e così via, consultate Attivazione dei tag di allocazione dei costi definiti dall'utente nella documenta zione. AWS Billing	DevOps ingegnere
Usa più ambienti.	Il GitHub repository fornisce una cartella di `development` ambiente. È inoltre possibile aggiungere i propri ambienti nella cartella environment. Per aggiungere un ambiente, copia la `development` cartella in una nuova cartella (ad esempio, `prod` `ostaging`) sotto`environme` `nts` . È quindi possibile aggiornare il `terraform` `.tfvars` file con il nuovo valore.	DevOps ingegnere

Implementa la landing zone

Operazione	Description	Competenze richieste
Inizializza la directory di lavoro di Terraform.	Per inizializzare la directory di lavoro e scaricare i pacchetti necessari, esegui il comando: `terraform init`	DevOps ingegnere
Crea un piano di esecuzione.	Per creare un piano di esecuzione e visualizzare le modifiche che Terraform apporterà alla tua infrastru ttura, esegui il comando: `terraform plan`	DevOps ingegnere
Implementa le modifiche.	Per implementare le modifiche all'infrastruttura come descritto nel codice, esegui il comando: `terraform apply`	DevOps ingegnere
Convalida la distribuzione.	Convalida i componenti che Terraform ha creato o modificato nella tua infrastru ttura. Per testare la configurazione, esegui il provisioning di una risorsa di calcolo (ad esempio, un' EC2 istanza o una AWS Lambda funzione Amazon) all'interno o collegata al VPC.	DevOps ingegnere, sviluppat ore di app

Rimuovi risorse

Operazione	Description	Competenze richieste
Elimina.	Al termine del test, esegui il seguente comando per distruggere le risorse distribuite da Terraform nella tua infrastruttura: `terraform destroy`	DevOps ingegnere

Risorse correlate

Scoperta e valutazione

- Suggerimenti amministrativi per la configurazione delle landing zone (AWS Control Tower documentazione)
- Aspettative per la configurazione delle landing zone (AWS Control Tower documentazione)
- Migliori pratiche per gli aggiornamenti delle landing zone (AWS Control Tower documentazione)

Configurazione di MongoDB Atlas e degli ambienti AWS

- Ottenere MongoDB Atlas ()Marketplace AWS
- Memoria (documentazione MongoDB Atlas)
- Esempio di dimensionamento con set di dati di esempio Atlas (documentazione MongoDB Atlas)
- Esempio di dimensionamento per applicazioni mobili (documentazione MongoDB Atlas)
- Traffico di rete (documentazione MongoDB Atlas)
- Auto-scaling del cluster (documentazione MongoDB Atlas)
- Modello di dimensionamento Atlas (documentazione MongoDB Atlas)
- Configurare una connessione peering di rete (documentazione MongoDB Atlas)
- Endpoint privati in Atlas (documentazione MongoDB Atlas)
- Crittografia a livello di campo lato client (documentazione del database MongoDB)
- Crittografia automatica (documentazione del database MongoDB)

- Selezione di un livello di cluster (documentazione MongoDB Atlas)

Implementazione della landing zone

- Terraform on AWS (CI/CD per reti 5G su white paper) AWS
- Atlante MongoDB con Terraform (documentazione MongoDB)

Configura i log di flusso VPC per la centralizzazione su Account AWS

Benjamin Morris e Aman Kaur Gandhi, Amazon Web Services

Riepilogo

In un cloud privato AWS virtuale (VPC), la funzionalità VPC Flow Logs può fornire dati utili per la risoluzione dei problemi operativi e di sicurezza. Tuttavia, esistono limitazioni all'utilizzo dei log di flusso VPC in un ambiente con più account. In particolare, i log di flusso tra account di Amazon CloudWatch Logs non sono supportati. Puoi invece centralizzare i log configurando un bucket Amazon Simple Storage Service (Amazon S3) con la policy dei bucket appropriata.

> **Note**
>
> Questo modello illustra i requisiti per l'invio dei log di flusso a una posizione centralizzata. Tuttavia, se desideri che i log siano disponibili anche localmente negli account dei membri, puoi creare più log di flusso per ogni VPC. Gli utenti che non hanno accesso all'account Log Archive possono visualizzare i registri del traffico per la risoluzione dei problemi. In alternativa, puoi configurare un singolo log di flusso per ogni VPC che invia i log a Logs. CloudWatch Puoi quindi utilizzare un filtro di abbonamento Amazon Data Firehose per inoltrare i log a un bucket S3. Per ulteriori informazioni, consulta la sezione Risorse correlate.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un' AWS Organizations organizzazione con un account utilizzato per centralizzare i registri (ad esempio, Log Archive)

Limitazioni

Se utilizzi la chiave gestita AWS Key Management Service (AWS KMS) `aws/s3` per crittografare il tuo bucket centrale, questo non riceverà i log da un altro account. Vedrai invece un codice di `Unsuccessful` errore 400 con un messaggio come quello `"LogDestination: <bucketName> is undeliverable"` che hai dato. `ResourceId` Questo perché le chiavi AWS gestite di un

account non possono essere condivise tra account. La soluzione consiste nell'utilizzare la crittografia gestita di Amazon S3 (SSE-S3) o una chiave gestita AWS KMS dal cliente che puoi condividere con gli account dei membri.

Architecture

Architettura Target

Nel diagramma seguente, vengono distribuiti due log di flusso per ogni VPC. Uno invia i log a un gruppo di Logs locale. CloudWatch L'altro invia i log a un bucket S3 in un account di registrazione centralizzato. La policy del bucket consente al servizio di consegna dei log di scrivere i log nel bucket.

> **Note**
>
> A partire da novembre 2023, AWS ora supporta la chiave di condizione aws: SourceOrg ID. Questa condizione ti consente di negare la scrittura nel bucket centralizzato per gli account esterni alla tua organizzazione. AWS Organizations

Automazione e scalabilità

Ogni VPC è configurato per inviare i log al bucket S3 nell'account di registrazione centrale. Utilizza una delle seguenti soluzioni di automazione per garantire che i log di flusso siano configurati in modo appropriato:

- CloudFormation StackSets
- AWS Control Tower Account Factory per Terraform (AFT)
- Qualsiasi AWS Config regola con correzione

Tools (Strumenti)

Strumenti

- Amazon CloudWatch Logs ti aiuta a centralizzare i log di tutti i tuoi sistemi e applicazioni, Servizi AWS così puoi monitorarli e archiviarli in modo sicuro.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS. Questo modello utilizza la funzionalità VPC Flow Logs per acquisire informazioni sul traffico IP in entrata e in uscita dalle interfacce di rete del tuo VPC.

Best practice

L'utilizzo dell'infrastruttura come codice (IaC) può semplificare notevolmente il processo di implementazione dei VPC Flow Logs. L'astrazione delle definizioni di distribuzione VPC per includere un costrutto di risorse per i log di flusso consentirà di implementare automaticamente VPCs i log di flusso. Questo è dimostrato nella prossima sezione.

Registri di flusso centralizzati

Sintassi di esempio per aggiungere log di flusso centralizzati a un modulo VPC in HashiCorp Terraform: questo codice crea un log di flusso che invia i log da un VPC a un bucket S3 centralizzato. Nota che questo modello non copre la creazione del bucket S3. Per le istruzioni sulle politiche relative ai bucket consigliate, consulta la sezione Informazioni aggiuntive.

```
variable "vpc_id" { type = string }
locals { custom_log_format_v5 = "$${version} $${account-id} $${interface-id} $
${srcaddr} $${dstaddr} $${srcport} $${dstport} $${protocol} $${packets} $${bytes}
 $${start} $${end} $${action} $${log-status} $${vpc-id} $${subnet-id} $${instance-
id} $${tcp-flags} $${type} $${pkt-srcaddr} $${pkt-dstaddr} $${region} $${az-id} $
${sublocation-type} $${sublocation-id} $${pkt-src-aws-service} $${pkt-dst-aws-service}
 $${flow-direction} $${traffic-path}" }
resource "aws_flow_log" "centralized_flow_log" {
  log_destination      = "arn:aws:s3:::centralized-vpc-flow-logs-
<log_archive_account_id>" # Optionally, a prefix can be added after the ARN.
  log_destination_type = "s3"
  traffic_type         = "ALL"
  vpc_id               = var.vpc_id
  log_format           = local.custom_log_format_v5 # If you want fields from VPC Flow
 Logs v3+, you will need to create a custom log format.
}
```

Per ulteriori informazioni sul formato di log personalizzato, consulta la documentazione di Amazon VPC.

Registri di flusso locali

Sintassi di esempio per aggiungere registri di flusso locali a un modulo VPC in Terraform con le autorizzazioni richieste: questo codice crea un log di flusso che invia i log da un VPC a un gruppo di log locale. CloudWatch

```
data "aws_region" "current" {}
variable "vpc_id" { type = string }
resource "aws_iam_role" "local_flow_log_role" {
  name = "flow-logs-policy-${var.vpc_id}"
  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [{
      "Effect": "Allow",
      "Principal": {"Service": "vpc-flow-logs.amazonaws.com"},
      "Action": "sts:AssumeRole"
  }]
}
EOF
}
resource "aws_iam_role_policy" "logs_permissions" {
  name = "flow-logs-policy-${var.vpc_id}"
  role = aws_iam_role.local_flow_log_role.id
  policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [{
      "Action": ["logs:CreateLog*", "logs:PutLogEvents", "logs:DescribeLog*",
 "logs:DeleteLogDelivery"],
      "Effect": "Allow",
      "Resource": "arn:aws:logs:${data.aws_region.current.name}:*:log-group:vpc-flow-
logs*"
  }]
}
EOF
}
resource "aws_cloudwatch_log_group" "local_flow_logs" {
  name              = "vpc-flow-logs/${var.vpc_id}"
  retention_in_days = 30
}
resource "aws_flow_log" "local_flow_log" {
  iam_role_arn    = aws_iam_role.local_flow_log_role.arn
```

```
  log_destination = aws_cloudwatch_log_group.local_flow_logs.arn
  traffic_type    = "ALL"
  vpc_id          = var.vpc_id
}
```

Epiche

Implementa l'infrastruttura VPC Flow Logs

Operazione	Description	Competenze richieste
Determina la strategia di crittografia e crea la policy per il bucket S3 centrale.	Il bucket centrale non supporta la AWS KMS `aws/s3` chiave, quindi è necessario utilizzare SSE-S3 o una chiave gestita dal cliente. AWS KMS Se si utilizza una AWS KMS chiave, la politica chiave deve consentire agli account dei membri di utilizzare la chiave.	Conformità
Crea il bucket centrale per i log di flusso.	Crea il bucket centrale a cui verranno inviati i log di flusso e applica la strategia di crittografia scelta nel passaggio precedente. Dovrebbe trovarsi in un Log Archive o in un account con scopi simili. Ottieni la policy sui bucket dalla sezione Informazioni aggiuntive e applicala al tuo bucket centrale dopo aver aggiornato i segnaposto con i valori specifici dell'ambiente.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Configura VPC Flow Logs per inviare i log al bucket di log di flusso centrale.	Aggiungi i log di flusso a ogni VPC da cui desideri raccoglie re dati. Il modo più scalabile per farlo è utilizzare strumenti IAc come AFT o. AWS Cloud Development Kit (AWS CDK) Ad esempio, puoi creare un modulo Terraform che distribui sce un VPC insieme a un log di flusso. Se necessari o, aggiungi i log di flusso manualmente.	Amministratore di rete
Configura i log di flusso VPC per l'invio ai log locali. CloudWatch	(Facoltativo) Se desideri che i log di flusso siano visibili negli account in cui vengono generati, crea un altro log di flusso per inviare i dati ai CloudWatch log nell'account locale. In alternativa, puoi inviare i dati a un bucket S3 specifico dell'account nell'acco unt locale.	Informazioni generali su AWS

Risorse correlate

- Come facilitare l'analisi dei dati e soddisfare i requisiti di sicurezza utilizzando dati di log di flusso centralizzati (post AWS sul blog)
- Come abilitare automaticamente i log di flusso VPC utilizzando AWS Config le regole (AWS post sul blog)

Informazioni aggiuntive

Politica Bucket

Questo esempio di policy sui bucket può essere applicato al bucket S3 centrale per i log di flusso, dopo aver aggiunto i valori per i nomi dei segnaposto.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AWSLogDeliveryWrite",
            "Effect": "Allow",
            "Principal": {
                "Service": "delivery.logs.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::<BUCKET_NAME>/*",
            "Condition": {
                "StringEquals": {
                    "s3:x-amz-acl": "bucket-owner-full-control",
                    "aws:SourceOrgID": "<ORG_ID>"
                }
            }
        },
        {
            "Sid": "AWSLogDeliveryCheck",
            "Effect": "Allow",
            "Principal": {
                "Service": "delivery.logs.amazonaws.com"
            },
            "Action": "s3:GetBucketAcl",
            "Resource": "arn:aws:s3:::<BUCKET_NAME>",
            "Condition": {
                "StringEquals": {
                    "aws:SourceOrgID": "<ORG_ID>"
                }
            }
        },
        {
            "Sid": "DenyUnencryptedTraffic",
            "Effect": "Deny",
            "Principal": {
                "AWS": "*"
            },
            "Action": "s3:*",
            "Resource": [
```

```
                "arn:aws:s3:::<BUCKET_NAME>/*",
                "arn:aws:s3:::<BUCKET_NAME>"
            ],
            "Condition": {
                "Bool": {
                    "aws:SecureTransport": "false"
                }
            }
        }
    ]
}
```

Gestisci i set di AWS autorizzazioni in modo dinamico utilizzando Terraform

Vinicius Elias e Marcos Vinicius Pinto Jordão, Amazon Web Services

Riepilogo

AWS IAM Identity Center migliora AWS Identity and Access Management (IAM) fornendo un hub centralizzato per la gestione dell'accesso Single Sign-On alle applicazioni cloud. Account AWS Tuttavia, la gestione manuale dei set di autorizzazioni di IAM Identity Center può diventare sempre più complessa e soggetta a errori man mano che l'organizzazione cresce. Questa complessità può portare a potenziali lacune di sicurezza e sovraccarichi amministrativi.

Questa soluzione consente di gestire i set di autorizzazioni tramite Infrastructure as Code (IaC) utilizzando una pipeline di integrazione e distribuzione continua (CI/CD) creata con tecnologia nativa. Servizi AWS Consente una perfetta integrazione del meccanismo di assegnazione del set di autorizzazioni con gli eventi del AWS Control Tower ciclo di vita o un ambiente Account Factory for Terraform (AFT). Questo approccio fornisce configurazioni di identità dinamiche sia per quelle nuove che per quelle esistenti. Account AWS

EventBridge Le regole di Amazon monitorano Account AWS la creazione e gli aggiornamenti, il che aiuta le configurazioni delle identità a rimanere sincronizzate con la struttura organizzativa. Dopo aver creato o aggiornato gli account in AWS Control Tower o AFT, viene attivata la pipeline. Valuta un set di file JSON con definizioni di set di autorizzazioni e regole di assegnazione. Quindi la pipeline applica e sincronizza le impostazioni su tutti gli account.

Questa funzionalità fornisce i seguenti vantaggi:

- Coerenza: elimina la deriva manuale della configurazione all'interno dell'organizzazione AWS
- Verificabilità: mantiene una cronologia completa di tutte le modifiche alla gestione delle identità
- Scalabilità: applica automaticamente le configurazioni man mano che l'ambiente cresce AWS
- Sicurezza: riduce l'errore umano nell'assegnazione delle autorizzazioni
- Conformità: facilita il rispetto dei requisiti normativi attraverso modifiche documentate e regole di assegnazione

Prerequisiti e limitazioni

- Un ambiente multi-account con AWS Control Tower e configurato. AWS Organizations Facoltativamente, puoi usare AFT con. AWS Control Tower
- Un amministratore delegato di IAM Identity Center Account AWS a ricevere la soluzione. Per ulteriori informazioni, consulta Amministrazione delegata nella documentazione di IAM Identity Center.
- Un repository del sistema di controllo della versione (VCS) per gestire il codice principale. Per un esempio, consultate il repository della soluzione. GitHub
- AWS Risorse necessarie per la gestione del backend Terraform, come un bucket Amazon Simple Storage Service (Amazon S3) e una tabella Amazon DynamoDB.

Limitazioni

- La pipeline utilizza AWS risorse native e Terraform open source. La pipeline non è predisposta per effettuare chiamate verso ecosistemi di terze parti.
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per informazioni sulla disponibilità per regione, consulta AWS Servizi per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Architecture

Il diagramma seguente mostra i componenti e il flusso di lavoro di questo modello.

AWS Control Tower flusso di eventi

La soluzione inizia con l'integrazione degli eventi provenienti da entrambi AWS Control Tower o da AFT. La scelta tra l'uno o l'altro servizio viene effettuata al momento dell'implementazione tramite una definizione variabile. Indipendentemente dal metodo utilizzato, la pipeline viene attivata ogni volta che un account viene creato o aggiornato. La pipeline riconcilia le politiche archiviate nell'archivio di gestione dei set di autorizzazioni.

Di seguito sono riportati gli eventi del ciclo di vita: AWS Control Tower

- `CreateManagedAccount`— Quando viene creato un nuovo account
- `UpdateManagedAccount`— Quando viene aggiornato un account esistente

Routing degli eventi

EventBridge funge da servizio centrale di elaborazione degli eventi, registrando gli eventi generati nell'account. AWS Control Tower Quando si verificano degli eventi, li indirizza EventBridge in modo intelligente a un bus di eventi centralizzato nell'account della soluzione. AWS Control Tower gli eventi del ciclo di vita seguono schemi di routing distinti. Se AFT è definito come origine dell'evento, l'account di gestione AFT gestisce gli eventi anziché l'account. AWS Control Tower Questa architettura basata sugli eventi consente risposte automatiche ai cambiamenti organizzativi senza intervento manuale.

Processo di integrazione AFT

Quando gli eventi AWS Control Tower del ciclo di vita raggiungono l'account di gestione AFT, attivano automaticamente più processi a valle intrinseci ad AFT. Una volta completato il flusso di lavoro di personalizzazione dell'account AFT, pubblica un messaggio nell'argomento dedicato Amazon Simple Notification Service (`aft-notifications`Amazon SNS). Questo argomento attiva la `aft-new-account-forward-event` AWS Lambda funzione implementata da questa soluzione. La funzione Lambda invia l'evento al bus degli eventi dell'account della soluzione, dove viene utilizzato per avviare la pipeline.

Infrastruttura come pipeline di codice

La pipeline di soluzioni funziona come un meccanismo di implementazione completamente automatizzato. Il AWS CodePipeline servizio monitora continuamente l'archivio per rilevare eventuali modifiche. Dopo aver rilevato nuovi commit, avvia automaticamente il flusso di lavoro di distribuzione e avvia un processo sequenziale che include fasi di convalida ed esecuzione. Il sistema esegue `plan` le operazioni Terraform per identificare le modifiche proposte, seguite dai `apply` comandi Terraform per implementare tali modifiche nell'ambiente. AWS In particolare, la pipeline funziona senza cancelli di approvazione manuale. Questo approccio consente una rapida implementazione delle modifiche all'infrastruttura mantenendo al contempo la verificabilità tramite i registri della pipeline e i file di stato Terraform.

La pipeline sfrutta le funzionalità di Terraform AWS CodeBuild per eseguire le operazioni di Terraform in un ambiente controllato con le autorizzazioni appropriate. Attraverso questo approccio IaC, la pipeline può eseguire operazioni complete di gestione delle autorizzazioni, tra cui:

- Creare nuovi set di autorizzazioni.
- Aggiorna i set di autorizzazioni esistenti.
- Rimuovere i set di autorizzazioni non necessari.

- Gestisci l'assegnazione di queste autorizzazioni tra account e gruppi all'interno delle AWS organizzazioni.

Per mantenere la coerenza dell'infrastruttura e prevenire modifiche in conflitto, la soluzione implementa il sistema di gestione dello stato di backend Terraform utilizzando un bucket Amazon S3 e una tabella Amazon DynamoDB dedicata. Questo approccio fornisce una posizione di archiviazione persistente per i file di stato Terraform e meccanismi di blocco dello stato per impedire modifiche simultanee alle stesse risorse.

Il codice Terraform principale utilizza il modulo Terraform ufficiale. AWS `permission-sets` Questo modulo può gestire dinamicamente i set di autorizzazioni in IAM Identity Center, sulla base di modelli di set di autorizzazioni.

Gestione del controllo del codice sorgente

I modelli di set di autorizzazioni (file JSON) risiedono in un sistema di controllo delle versioni esterno, ad esempio GitHub, che fornisce un archivio centralizzato per le configurazioni di gestione delle identità. Questo approccio stabilisce un'unica fonte di verità per le definizioni dei set di autorizzazioni, consentendo al contempo lo sviluppo collaborativo attraverso pratiche standard di revisione del codice. Gli utenti autorizzati possono apportare modifiche a questi modelli seguendo i processi di gestione delle modifiche organizzative. Questi commit fungono da trigger principale per la pipeline di distribuzione automatizzata, che avvia il processo di aggiornamento dell'infrastruttura.

Per un esempio di come configurare i set di autorizzazioni utilizzando il file JSON nel repository, consulta Informazioni aggiuntive.

Tools (Strumenti)

Servizi AWS

- AWS CodeBuildè un servizio di compilazione completamente gestito che consente di compilare il codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS CodeConnectionsconsente a AWS risorse e servizi, ad esempio, di connettersi a repository di codice esterni CodePipeline, come. GitHub
- AWS CodePipelineaiuta a modellare e configurare rapidamente le diverse fasi di un rilascio software e ad automatizzare i passaggi necessari per rilasciare le modifiche software in modo continuo.

- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando.
- AWS Control Towerti aiuta a configurare e gestire un ambiente con AWS più account, seguendo le migliori pratiche prescrittive.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altro modo. Account AWS
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS IAM Identity Centerti aiuta a gestire centralmente l'accesso Single Sign-On (SSO) a tutte le tue applicazioni e a quelle sul cloud. Account AWS
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS Organizationsè un servizio di gestione degli account che ti aiuta a consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail. Abilita le notifiche push per gli eventi di gestione degli account, assicurando che le parti interessate siano informate di importanti modifiche o azioni all'interno del sistema.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri strumenti

- Terraform è uno strumento di infrastruttura come codice (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.

Deposito di codici

Il codice per questo pattern è disponibile nell'organizzazione AWS Samples del repository sample-terraform-aws-permission-sets-pipeline. GitHub

Best practice

- Aggiungi sempre le versioni dei moduli e dei provider Terraform utilizzati per eseguire il codice in produzione.
- Utilizza uno strumento di analisi statico del codice, come Checkov, per scansionare il codice e quindi risolvere i problemi di sicurezza.
- Segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.

Epiche

Crea i prerequisiti (opzionale)

Operazione	Description	Competenze richieste
Crea risorse di backend Terraform.	Se non hai ancora creato le tue AWS risorse di backend Terraform, utilizza i seguenti passaggi per creare un bucket Amazon S3 `s3-tf-backend-{ACCOUNT_ID}` () e una tabella DynamoDB (). `ddb-tf-backend` 1. Accedi al luogo in Account AWS cui distribuirai la soluzione e apri nella home page. AWS CloudShell AWS Control Tower Regione AWS 2. Esegui i seguenti comandi, sostituendo il segnaposto `{ACCOUNT_ID}` con il tuo ID: Account AWS	Amministratore AWS

Operazione	Description	Competenze richieste

	`aws s3api create-bucket --bucket s3-tf-backend-{ACCOUNT_ID}`
	`aws s3api put-bucket-versioning --bucket s3-tf-backend-{ACCOUNT_ID} --versioning-configuration Status=Enabled`
	`aws dynamodb create-table --table-name ddb-tf-backend --attribute-definitions AttributeName=LockID,AttributeType=S --key-schema AttributeName=LockID,KeyType=HASH --provisioned-throughput ReadCapacityUnits=1,WriteCapacityUnits=1`	

Operazione	Description	Competenze richieste
Crea un ruolo per più account.	È necessario fornire un ruolo IAM tra account nella configurazione del provider `event-source-account` Terraform AWS . Se non hai ancora creato questo ruolo, utilizza i seguenti passaggi per crearlo: 1. Accedi a Account AWS quella che sarà la fonte del tuo evento (account di AWS Control Tower gestione o account AFT) e apri AWS CloudShell. 2. Esegui il comando seguente, sostituendo il segnaposto `{ACCOUNT_ID}` con l' Account AWS ID che stai utilizzando per questa soluzione, non con l'ID attuale Account AWS : `aws iam create-role \ --role-name CrossAccountRole \ --assume-role-policy-document '{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": {`	Amministratore AWS

Operazione	Description	Competenze richieste

	<pre>"AWS": "arn:aws:iam::{ACCOUNT_ID}:root"
	},
	
	"Action": "sts:AssumeRole"
	}
	]
	}'</pre>
	1. Controlla l'output del comando e copia il ruolo Amazon Resource Name (ARN) (ad esempio,`arn:aws:iam::111122223333:role/CrossAccountRole`).
	2. Per allegare una policy IAM al ruolo, esegui il seguente comando:
	
	<pre>aws iam attach-role-policy \
	--role-name CrossAccountRole \
	--policy-arn arn:aws:iam::aws:policy/AdministratorAccess</pre>
	Questo esempio utilizza la policy IAM AWS gestita AdministratorAccess. Se	

Operazione	Description	Competenze richieste
	preferisci, puoi utilizzare una politica più specifica.	

Preparare l'archivio dei set di autorizzazioni

Operazione	Description	Competenze richieste
Crea un repository dedicato.	Questa attività presuppone che tu stia utilizzando. GitHub Crea un repository dedicato per archiviare il codice Terraform principale e i file JSON del modello di set di autorizzazioni.	DevOps ingegnere
Prepara il codice del set di autorizzazioni.	Per informazioni su come strutturare i seguenti file, consultate il codice di esempio nel repository della soluzione: — main.tf – output.tf – providers.jinja — modelli Copia il contenuto, conserva i `providers.jinja` valori e apporta le modifiche necessarie agli altri file. Ad esempio, aggiungi i file modello del set di autorizzazioni `templates` o aggiungi la versione del `aws-ia/permission-`	DevOps ingegnere

Operazione	Description	Competenze richieste
	`sets/aws` modulo nel `main.tf` file.	
Effettua le tue modifiche.	Conferma e invia le modifiche al repository che hai creato in precedenza. Salva il nome del repository e la sua GitHub organizzazione, ad esempio. `myorg/aws-ps-pipeline`	DevOps ingegnere

Prepara il codice di distribuzione

Operazione	Description	Competenze richieste
Scarica il contenuto.	Scarica (clona) il contenuto dal repository della soluzione.	DevOps ingegnere
Soddisfa le variabili.	Crea un `terraform .tfvars` file e aggiungi le seguenti variabili necessarie: • `repository_name` — Nome dell'archivio dei set di autorizzazioni creato in precedenza • `branch_name` — Nome del ramo del repository del set di autorizzazioni • `vcs_provider` — Il tuo provider VCS • `account_lifecycle_events_source` — La fonte degli eventi che	DevOps ingegnere

Operazione	Description	Competenze richieste

	innescano la pipeline (AWS Control Tower o AFT)
	
	`repository_name = "myorg/aws-ps-pipeline"`
	`branch_name = "main"`
	`vcs_provider = "github"`
	`account_lifecycle_events_source = "CT"`
	
	Per informazioni sulle opzioni variabili aggiuntive, consultate il file variables.tf nel repository di questo pattern. GitHub	

Operazione	Description	Competenze richieste
Regola la configurazione del backend Terraform.	Nel `backend.tf` file, sostituisci i segnaposto con i tuoi valori. Usa la AWS Control Tower home Regione AWS e fornisci i nomi del bucket Amazon S3 e della tabella DynamoDB creati in precedenza. `terraform { required_version = ">=1.6" backend "s3" { region = "{region}" bucket = "{bucket_name}" key = "terraform.tfstate" dynamodb_table = "{table_name}" encrypt = "true" } }` Se preferisci, puoi usare la tua configurazione di backend Terraform.	DevOps ingegnere

Operazione	Description	Competenze richieste

Regola la configurazione del provider Terraform.	Nel `providers.tf` file, sostituisci i segnaposto con le tue informazioni. Utilizza la AWS Control Tower home Region e fornisci l'ARN del ruolo IAM tra account precedentemente creato per il provider. `event-source-account`
	`provider "aws" {`
	`  region = "{region}"`
	`}`
	
	`provider "aws" {`
	`  alias  = "event-so`
	`urce-account"`
	`  region = "{region}"`
	`  assume_role {`
	`  role_arn = "{role_ar`
	`n}"`
	`  }`
	`}`
		DevOps ingegnere

Implementa la soluzione manualmente

Operazione	Description	Competenze richieste
Seleziona il Account AWS.	Ti consigliamo di implement are la soluzione nell'acco unt amministratore delegato di IAM Identity Center. Tuttavia, puoi anche distribui rla nell'account di gestione. AWS Organizations	Amministratore AWS

Operazione	Description	Competenze richieste
	Per accedere all'account selezionato nella stessa regione dell'istanza IAM Identity Center, utilizza il AWS CLI. Assicurati che il ruolo IAM che stai utilizzando sia autorizzato ad assumere il ruolo specificato per il `event-source-account` provider nei passaggi precedenti. Inoltre, questo ruolo deve avere accesso alle AWS risorse utilizzate nella configurazione del backend Terraform.	
Esegui Terraform manualmente.	Per inizializzare, pianificare e applicare le configurazioni, esegui i seguenti comandi Terraform nell'ordine mostrato: 1. `terraform init` 2. `terraform plan` 3. `terraform apply`	DevOps ingegnere
Il controllo dei risultati della distribuzione.	Nell'account amministratore delegato di IAM Identity Center, verifica che la `aws-ps-pipeline` pipeline sia stata creata. Verifica anche che esista una AWS CodeConnections connessione con lo stato In sospeso.	AWS DevOps

Operazione	Description	Competenze richieste
Completa la CodeConnections configurazione.	Per completare la CodeConnections configurazione, procedi nel seguente modo: 1. Vai a CodeConnections, seleziona la connessione che è stata creata e scegli Aggiorna connessione in sospeso. 2. Fornisci GitHub le tue credenziali, installa una nuova GitHub app o selezionane una esistente che abbia accesso all'archivio dei set di autorizzazioni e scegli Connect. La pipeline dovrebbe ora avere accesso all'archivio dei set di autorizzazioni. Per istruzioni dettagliate, consulta Aggiornare una connessione in sospeso nella documentazione della console Developer Tools.	AWS DevOps

Scegli un flusso di esecuzione della pipeline per testare la soluzione

Operazione	Description	Competenze richieste
Esegui la pipeline AWS Control Tower tramite aggiornamenti AFT.	Dopo la creazione o la modifica di un account utilizzando AWS Control	Amministratore AWS

Operazione	Description	Competenze richieste
	Tower o AFT (a seconda del tipo di eventi del ciclo di vita scelti), viene avviata la pipeline.	
Esegui la pipeline modificando il codice.	Dopo aver modificato il codice e averlo inviato al `main` ramo, viene avviata la pipeline.	AWS DevOps
Esegui la pipeline manualmente.	Per avviare la pipeline manualmente, utilizzate la funzione Release change in. AWS CodePipeline	AWS DevOps

Risoluzione dei problemi

Problema	Soluzione
Accesso negato	Verifica di disporre delle autorizzazioni necessarie per distribuire la soluzione.
CodeConnections problemi	• Verifica che lo stato della connessione sia Disponibile anziché In sospeso. • Verifica che la CodeConnections configurazione sia completata correttamente con GitHub le credenziali. Per ulteriori informazioni, consulta Aggiornare una connessione in sospeso nella documentazione della console Developer Tools. • Assicurati che l' GitHub app disponga delle autorizzazioni di accesso adeguate all'archivio dei set di autorizzazioni.
Problemi di esecuzione della pipeline	• Controlla i CloudWatch log di Amazon per eventuali errori di esecuzione della pipeline.

Problema	Soluzione
	• Assicurati che la delegazione di IAM Identity Center sia configurata correttamente. In caso contrario, può causare errori nella pipeline durante la lettura Account AWS dei metadati.
Problemi di distribuzione dei set di autorizzazioni	• Verifica che la sintassi JSON nei file modello del set di autorizzazioni sia corretta. • Verifica che esistano le policy gestite da IAM a cui si fa riferimento nei modelli. • Verifica se i gruppi e gli utenti specificati nelle assegnazioni esistono in IAM Identity Center.

Risorse correlate

Servizio AWS documentazione

- AWS IAM Identity Center Guida per l'utente
- Gestione Account AWS con set di autorizzazioni (documentazione IAM Identity Center)

Altre risorse

- AWS Modulo Permission Sets (Terraform)

Informazioni aggiuntive

File JSON con set di autorizzazioni di esempio

L'esempio seguente mostra come configurare un set di autorizzazioni utilizzando il file JSON nel repository:

```
{
  "Name": "ps-billing", // Permission set identifier
  "Comment": "Sample permission set for billing access", // Comment to document the
 purpose of the permission set
  "Description": "Billing access in AWS", // Detailed description
```

```
  "SessionDuration": "PT4H", // Session duration = 4 hours (ISO 8601 format)
  "ManagedPolicies": [ // List of AWS IAM managed policies
    "arn:aws:iam::aws:policy/job-function/Billing",
    "arn:aws:iam::aws:policy/job-function/SupportUser",
    "arn:aws:iam::aws:policy/AWSSupportAccess",
    "arn:aws:iam::aws:policy/job-function/ViewOnlyAccess"
  ],
  "CustomerPolicies": [], // References to IAM policies previously created
  "CustomPolicy": {}, // Inline IAM policy defined directly in the permission set
  "PermissionBoundary": {  // AWS or customer managed IAM policy to be used as boundary
    "ManagedPolicy": "",
    "CustomerPolicy": ""
  },
  "Assignments": [ // Define the assignment rules
    {
      "all_accounts": true, // Apply to ALL active AWS accounts in organization
      "principal": "G_BILLING_USERS", // Group/user name in Identity Center
      "type": "GROUP", // Can be "GROUP" or "USER"
      "account_id": [], // List of AWS account ID (empty since all_accounts=true)
      "account_ou": [], // List of AWS Organizational Unit IDs with target AWS accounts
      "account_tag": [] // List of tags (key:value) to match AWS Organization accounts
 tags
    }
  ]
}
```

Per ulteriori informazioni, consulta lo schema JSON nella documentazione del modulo AWS Permission Sets sul sito Web Terraform.

Suggerimenti

- Puoi utilizzare i blocchi di importazione Terraform per importare un set di autorizzazioni esistente nella soluzione.
- È possibile utilizzare AFT per implementare la pipeline del set di AWS autorizzazioni in un account delegato. Per ulteriori informazioni, consulta AFT Blueprints.

Etichetta automaticamente gli allegati Transit Gateway utilizzando AWS Organizations

Richard Milner-Watts, Haris Bin Ayub e John Capps, Amazon Web Services

Riepilogo

Su Amazon Web Services (AWS), puoi utilizzarlo AWS Resource Access Managerper condividere AWS Transit Gatewayoltre Account AWS i confini. Tuttavia, quando si creano allegati Transit Gateway oltre i limiti dell'account, gli allegati vengono creati senza un tag Name. Ciò può rendere l'identificazione degli allegati dispendiosa in termini di tempo.

Questa soluzione fornisce un meccanismo automatizzato per raccogliere informazioni su ogni allegato Transit Gateway per gli account all'interno di un'organizzazione gestita da AWS Organizations. Il processo include la ricerca dell'intervallo Classless Inter-Domain Routing (CIDR) dalla tabella di routing Transit Gateway. La soluzione applica quindi un tag Name sotto forma di all'allegato all'interno dell'`<CIDR-range>-<AccountName>`account che contiene il gateway di transito.

Questa soluzione può essere utilizzata insieme a una soluzione come il Serverless Transit Network Orchestrator della Solutions Library. AWS Serverless Transit Network Orchestrator consente la creazione automatizzata di allegati Transit Gateway Gateway su larga scala.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un' AWS Organizations organizzazione che contiene tutti gli account correlati
- Accesso all'account di gestione dell'organizzazione, nella directory principale dell'organizzazione, per creare il ruolo richiesto AWS Identity and Access Management (IAM)
- Un account membro della rete condivisa contenente uno o più gateway di transito condivisi con l'organizzazione e dotati di allegati

Architecture

La seguente schermata Console di gestione AWS mostra esempi di allegati Transit Gateway senza tag Name associato e due allegati Transit Gateway con tag Name generati da questa soluzione. La struttura del tag Name generato è. `<CIDR-range>-<AccountName>`

Questa soluzione utilizza un AWS Step Functionsflusso AWS CloudFormationdi lavoro che gestisce la creazione di tag Transit Gateway Name su tutte le Regioni AWS configurazioni. Il flusso di lavoro richiama AWS Lambdafunzioni che eseguono le attività sottostanti.

Dopo che la soluzione ha ottenuto i nomi degli account da AWS Organizations, la macchina a stati Step Functions ottiene tutti gli allegati Transit Gateway IDs. Questi vengono elaborati in parallelo per regione. Questa elaborazione include la ricerca dell'intervallo CIDR per ogni allegato. L'intervallo CIDR si ottiene cercando nelle tabelle di routing Transit Gateway all'interno della regione un ID allegato Transit Gateway corrispondente. Se tutte le informazioni richieste sono disponibili, la soluzione applica un tag Name all'allegato. La soluzione non sovrascriverà alcun tag Name esistente.

La soluzione viene eseguita secondo una pianificazione controllata da un EventBridge evento Amazon. L'evento avvia la soluzione ogni giorno alle 06:00 UTC.

Stack tecnologico Target

- Amazon EventBridge
- AWS Lambda
- AWS Organizations
- AWS Transit Gateway
- Amazon Virtual Private Cloud (Amazon VPC) (Amazon VPC)
- AWS X-Ray

Architettura Target

L'architettura della soluzione e il flusso di lavoro sono illustrati nel diagramma seguente.

1. L'evento pianificato avvia la regola.
2. La EventBridge regola avvia la macchina a stati Step Functions.

3. La macchina a stati richiama la funzione Lambda`tgw-tagger-organizations-account-query`.
4. La funzione `tgw-tagger-organizations-account-query` Lambda assume il ruolo nell'account di gestione dell'organizzazione.
5. La funzione `tgw-tagger-organizations-account-query` Lambda richiama l'API Organizations per restituire Account AWS i metadati.
6. La macchina a stati richiama la funzione Lambda`tgw-tagger-attachment-query`.
7. Per ogni regione, in parallelo, la macchina a stati richiama la funzione `tgw-tagger-rtb-query` Lambda per leggere l'intervallo CIDR per ogni allegato.
8. Per ogni regione, in parallelo, la macchina a stati richiama la funzione Lambda `tgw-tagger-attachment-tagger`.
9. I name tag vengono creati per gli allegati Transit Gateway nell'account Shared Networking.

Automazione e scalabilità

La soluzione elabora ogni regione in parallelo per ridurre la durata totale dell'esecuzione.

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationfornisce un modo per modellare una raccolta di risorse correlate AWS e di terze parti, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita, trattando l'infrastruttura come codice.
- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue risorse AWS e delle applicazioni su cui esegui AWS in tempo reale.
- Amazon EventBridge è un servizio di bus eventi senza server che puoi utilizzare per connettere le tue applicazioni con dati provenienti da una varietà di fonti. EventBridge riceve un evento, un indicatore di un cambiamento nell'ambiente e applica una regola per indirizzare l'evento verso un obiettivo. Le regole abbinano gli eventi agli obiettivi in base alla struttura dell'evento, chiamata pattern di evento, o a una pianificazione.
- AWS Lambdaè un servizio di elaborazione che supporta l'esecuzione di codice senza fornire o gestire server. Lambda esegue il codice solo quando necessario e si ridimensiona automaticamente, da poche richieste al giorno a migliaia al secondo. I costi saranno calcolati in base al tempo di elaborazione effettivo. Quando il codice non è in esecuzione non viene addebitato alcun costo.

- AWS Organizationsti aiuta a gestire e governare centralmente il tuo ambiente man mano che cresci e scalerai le tue risorse. AWS Utilizzando Organizations, puoi creare nuove risorse Account AWS e allocarle in modo programmatico, raggruppare account per organizzare i flussi di lavoro, applicare politiche agli account o ai gruppi per la governance e semplificare la fatturazione utilizzando un unico metodo di pagamento per tutti i tuoi account.
- AWS Step Functionsè un servizio di flusso di lavoro visivo a basso codice utilizzato per orchestrare Servizi AWS, automatizzare i processi aziendali e creare applicazioni serverless. I flussi di lavoro gestiscono gli errori, i nuovi tentativi, la parallelizzazione, le integrazioni dei servizi e l'osservabilità, in modo che gli sviluppatori possano concentrarsi su logiche aziendali di maggior valore.
- AWS Transit Gatewayconnette e gestisce reti locali tramite un hub centrale. VPCs Ciò semplifica la rete e pone fine alle complesse relazioni di peering. Funziona come un router cloud, in modo che ogni nuova connessione venga effettuata una sola volta.
- Amazon Virtual Private Cloud (Amazon VPC) è un servizio per l'avvio di AWS risorse in una rete virtuale logicamente isolata da te definita.
- AWS X-Rayraccoglie dati sulle richieste servite dalla tua applicazione e fornisce strumenti che puoi utilizzare per visualizzare, filtrare e ottenere informazioni su tali dati per identificare problemi e opportunità di ottimizzazione.

Codice

Il codice sorgente di questa soluzione è disponibile nel GitHub repository Transit Gateway Attachment Tagger. Il repository include i seguenti file:

- `tgw-attachment-tagger-main-stack.yaml`crea tutte le risorse per supportare questa soluzione all'interno dell'account Shared Networking.
- `tgw-attachment-tagger-organizations-stack.yaml`crea un ruolo nell'account di gestione dell'organizzazione.

Epiche

Implementa lo stack di soluzioni principale

Operazione	Description	Competenze richieste
Raccogli le informazioni necessarie sui prerequisiti.	Per configurare l'accesso tra account dalla funzione	DevOps ingegnere

Operazione	Description	Competenze richieste
	Lambda all'API, è necessario AWS Organizations l'ID account per l'account di gestione dell'organizzazione. **Note** L'ordine in cui vengono creati i due CloudFormation stack è important e. È necessario prima distribuire le risorse nell'account di rete condivisa . Il ruolo nell'account di rete condivisa deve già esistere prima di distribuire le risorse nell'account di gestione dell'organizzazione. Per ulteriori informazioni, consulta la documentazione relativa ad AWS.	

Operazione	Description	Competenze richieste
Avvia il CloudFormation modello per lo stack di soluzioni principale.	Il modello per lo stack di soluzioni principale implementerà i ruoli IAM, il flusso di lavoro Step Functions, le funzioni Lambda e l'evento Amazon. CloudWatch Apri l'account Console di gestione AWS for the Shared Networking, quindi apri la console: &CFN. Crea lo stack utilizzando il `tgw-attachment-tagger-main-stack.yaml` modello e i seguenti valori: • Nome dello stack: -stack tgw-attachment-tagger-main • awsOrganizationsRootAccountId— ID dell'account di gestione dell'organizzazione • TGWRegionsparametro : Regioni AWS per la soluzione, immesso come stringa delimitata da virgole • TGWListparametro — Gateway di transito da IDs escludere dalla soluzione , inserito in una stringa delimitata da virgole	DevOps ingegnere

Operazione	Description	Competenze richieste
	Per ulteriori informazioni sul lancio di uno CloudFormation stack, consulta la documentazione.AWS	
Verifica che la soluzione sia stata avviata correttamente.	Attendi che lo CloudFormation stack raggiunga lo stato CREATE_COMPLETE. Questa operazione dovrebbe richiedere meno di un minuto. Apri la console Step Functions e verifica che sia stata creata una nuova macchina a stati con il nome tgw-attachment-tagger-state-machine.	DevOps ingegnere

Implementa lo stack AWS Organizations

Operazione	Description	Competenze richieste
Raccogli le informazioni necessarie sui prerequisiti.	Per configurare l'accesso tra account dalla funzione Lambda all'API AWS Organizations, è necessario l'ID account per l'account Shared Networking.	DevOps ingegnere
Avvia il CloudFormation modello per lo stack Organizations	Il modello per lo stack AWS Organizations implementerà il ruolo IAM nell'account di gestione dell'organizzazione. Accedi alla console AWS per l'account di gestione dell'organizzazione, quindi apri la	DevOps ingegnere

Operazione	Description	Competenze richieste
	CloudFormation console. Crea lo stack utilizzando il `tgw-attachment-tagger-organizations-stack.yaml` modello e i seguenti valori: • Nome dello stack: -stack tgw-attachment-tagger-organizations • NetworkingAccountId parametro — ID account per l'account di rete condivisa Per le altre opzioni di creazione dello stack, utilizzate le impostazioni predefinite.	
Verifica che la soluzione sia stata avviata correttamente.	Attendi che lo CloudFormation stack raggiunga lo stato CREATE_COMPLETE. Questa operazione dovrebbe richiedere meno di un minuto. Apri la console AWS Identity and Access Management (IAM) e verifica che sia stato creato un nuovo ruolo con il nome tgw-attachment-tagger-organization-query-role.	DevOps ingegnere

Verifica la soluzione

Operazione	Description	Competenze richieste
Avvia la macchina a stati.	Apri la console Step Functions per l'account Shared Networking e scegli Macchine a stati nel riquadro di navigazione. Seleziona la macchina a stati tgw-attachment-tagger-state-machine e scegli Avvia esecuzione. Poiché l'input di questa macchina a stati non viene utilizzato dalla soluzione, è possibile utilizzare il valore predefinito. `{ "Comment": "Insert your JSON here" }` Selezionare Start Execution (Avvia esecuzione).	DevOps ingegnere
Sorveglia la macchina statale fino al completamento.	Nella nuova pagina che si apre, puoi guardare la macchina a stati funzionar e. La durata dipenderà dal numero di allegati Transit Gateway da elaborare. In questa pagina è possibile esaminare ogni fase della	DevOps ingegnere

Operazione	Description	Competenze richieste
	macchina a stati. È possibile visualizzare le varie attività all'interno della macchina a stati e seguire i collegamenti ai CloudWatch registri delle funzioni Lambda. Per le attività eseguite in parallelo all'inter no della mappa, puoi utilizzar e l'elenco a discesa Indice per visualizzare le implement azioni specifiche per ogni regione.	
Verifica i tag degli allegati Transit Gateway.	Apri la console VPC per l'account Shared Networkin g e scegli Transit Gateway Attachments. Sulla console, viene fornito un tag Name per gli allegati che soddisfan o i criteri (l'allegato viene propagato a una tabella di routing Transit Gateway e il proprietario della risorsa è un membro dell'organizzazione).	DevOps ingegnere

Operazione	Description	Competenze richieste
Verifica l'inizio CloudWatch dell'evento.	Attendi l'inizio CloudWatch dell'evento. L'evento è previsto per le 06:00 UTC. Quindi apri la console Step Functions per l'account Shared Networking e scegli Macchine a stati nel riquadro di navigazione. Seleziona la macchina a stati tgw-attachment-tagger-state -macchina. Verifica che la soluzione sia stata eseguita alle 06:00 UTC.	DevOps ingegnere

Risorse correlate

- AWS Organizations
- AWS Resource Access Manager
- Orchestrator di rete di transito senza server
- Creazione di ruoli IAM
- Creazione di uno stack sulla console AWS CloudFormation

Altri modelli

- Automatizza le allocazioni IPv4 IPAM CIDR di Amazon VPC per nuove applicazioni utilizzando AFT Account AWS
- Gestisci i set di autorizzazioni per più account utilizzando Account Factory for Terraform
- Esegui il provisioning di AWS Service Catalog prodotti basati su AWS CloudFormation modelli utilizzando GitHub Actions
- Fornisci ruoli IAM con privilegi minimi implementando una soluzione di distribuzione automatica di ruoli

AI e apprendimento automatico

Argomenti

- Associa un AWS CodeCommit repository in uno Account AWS ad Amazon SageMaker AI Studio Classic in un altro account
- Estrai automaticamente i contenuti dai file PDF utilizzando Amazon Textract
- Crea un modello di previsione con avvio a freddo utilizzando DeepAR per le serie temporali in SageMaker Amazon AI Studio Lab
- Crea un MLOps flusso di lavoro usando Amazon SageMaker AI e Azure DevOps
- Configura la registrazione delle chiamate dei modelli in Amazon Bedrock utilizzando AWS CloudFormation
- Crea un'immagine di contenitore Docker personalizzata SageMaker e usala per l'addestramento dei modelli in AWS Step Functions
- Usa gli agenti Amazon Bedrock per automatizzare la creazione di controlli di accesso in Amazon EKS tramite istruzioni basate su testo
- Implementa uno use case RAG AWS utilizzando Terraform e Amazon Bedrock
- Implementa la logica di preelaborazione in un modello ML in un singolo endpoint utilizzando una pipeline di inferenza in Amazon SageMaker
- Implementa la convalida della sicurezza della codifica in tempo reale utilizzando un server MCP con Kiro e altri assistenti di codifica
- Sviluppa assistenti avanzati basati sull'intelligenza artificiale generativa utilizzando RAG e suggerimenti ReAct
- Sviluppa un assistente basato su chat completamente automatizzato utilizzando gli agenti e le knowledge base di Amazon Bedrock
- Documenta le conoscenze istituzionali utilizzando input vocali utilizzando Amazon Bedrock e Amazon Transcribe
- Genera consigli personalizzati e riclassificati con Amazon Personalize
- Semplifica i flussi di lavoro di machine learning dallo sviluppo locale agli esperimenti scalabili utilizzando SageMaker AI e Hydra
- Traduci il linguaggio naturale in query DSL for OpenSearch ed Elasticsearch
- Usa Amazon Q Developer come assistente di programmazione per aumentare la tua produttività

- Usa SageMaker Processing per l'ingegneria di funzionalità distribuite di set di dati ML su scala terabyte
- Visualizza i risultati AI/ML del modello utilizzando Flask e AWS Elastic Beanstalk
- Altri modelli

Associa un AWS CodeCommit repository in uno Account AWS ad Amazon SageMaker AI Studio Classic in un altro account

Laurens van der Maas e Aubrey Oosthuizen, Amazon Web Services

Riepilogo

Questo modello fornisce istruzioni e codice su come associare un AWS CodeCommit repository in uno Account AWS (Account A) con Amazon SageMaker AI Studio Classic in un altro Account AWS (Account B). Per configurare l'associazione, devi creare una policy e un ruolo AWS Identity and Access Management (IAM) nell'Account A e una policy IAM in linea nell'Account B. Quindi, usi uno script di shell per clonare il CodeCommit repository dall'Account A ad Amazon SageMaker AI Classic nell'Account B.

Prerequisiti e limitazioni

Prerequisiti

- Due Account AWS, uno contenente il CodeCommit repository e l'altro contenente un dominio SageMaker AI con un utente
- Dominio e utente SageMaker AI forniti, con accesso a Internet o accesso a CodeCommit e AWS Security Token Service (AWS STS) tramite endpoint di rete privata virtuale (VPC)
- Una conoscenza di base di IAM
- Una conoscenza di base di SageMaker AI Studio Classic
- Una conoscenza di base di Git e CodeCommit

Limitazioni

Questo modello si applica solo a SageMaker AI Studio Classic, non ad RStudio Amazon SageMaker AI.

Architecture

Stack tecnologico

- Amazon SageMaker AI
- Amazon SageMaker AI Studio Classic

- AWS CodeCommit
- AWS Identity and Access Management (IAM)
- Git

Architettura Target

Il diagramma seguente mostra un'architettura che associa un CodeCommit repository dall'Account A a SageMaker AI Studio Classic nell'Account B.

Il diagramma mostra il flusso di lavoro seguente:

1. Un utente assume il `MyCrossAccountRepositoryContributorRole` ruolo nell'Account A attraverso il `sts:AssumeRole` ruolo, mentre utilizza il ruolo di esecuzione SageMaker AI in SageMaker AI Studio Classic nell'Account B. Il ruolo assunto include le CodeCommit autorizzazioni per clonare e interagire con il repository specificato.
2. L'utente esegue i comandi Git dal terminale di sistema in SageMaker AI Studio Classic.

Automazione e scalabilità

Questo modello è costituito da passaggi manuali che possono essere automatizzati utilizzando AWS Cloud Development Kit (AWS CDK)AWS CloudFormation, o Terraform.

Tools (Strumenti)

Strumenti AWS

- Amazon SageMaker AI è un servizio di machine learning (ML) gestito che ti aiuta a creare e addestrare modelli di machine learning per poi distribuirli in un ambiente ospitato pronto per la produzione.
- Amazon SageMaker AI Studio Classic è un ambiente di sviluppo integrato (IDE) basato sul Web per l'apprendimento automatico che ti consente di creare, addestrare, eseguire il debug, distribuire e monitorare i tuoi modelli di apprendimento automatico.
- AWS CodeCommitè un servizio di controllo delle versioni che consente di archiviare e gestire in modo privato gli archivi Git, senza dover gestire il proprio sistema di controllo del codice sorgente.

  Avviso: non AWS CodeCommit è più disponibile per i nuovi clienti. I clienti esistenti di AWS CodeCommit possono continuare a utilizzare il servizio normalmente. Ulteriori informazioni

- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.

Altri strumenti

- Git è un sistema distribuito di controllo delle versioni per tenere traccia delle modifiche nel codice sorgente durante lo sviluppo del software.

Epiche

Crea una policy IAM e un ruolo IAM nell'Account A

Operazione	Description	Competenze richieste
Crea una policy IAM per l'accesso al repository nell'Account A.	1. Accedi Console di gestione AWS e apri la console IAM. 2. Nel pannello di navigazione, scegliere Policies (Policy) e Create Policy (Crea policy). 3. Scegli la scheda JSON. 4. Copia la dichiarazione di policy da Example IAM policy nella sezione Informazioni aggiuntive di questo pattern, quindi incolla la dichiarazione nell'editor JSON. Assicurati di sostituire tutti i valori segnaposto nella policy. 5. Scegli Avanti:Tag, quindi scegli Avanti:Revisione. 6. In Name (Nome), immettere un nome per la policy. Nota: in questo modello, la policy IAM viene	AWS DevOps

Operazione	Description	Competenze richieste
	chiamata`CrossAccountAccessForMySharedDemoRepo` , ma puoi scegliere il nome della policy che preferisci. 7. Scegli Crea policy. **Tip** È buona prassi limitare l'ambito delle policy IAM alle autorizzazioni minime richieste per il tuo caso d'uso.	

Operazione	Description	Competenze richieste
Crea un ruolo IAM per l'accesso al repository nell'Account A.	1. Nel pannello di navigazione della console IAM, scegli Ruoli e poi Crea ruolo. 2. Per il tipo di entità affidabile, seleziona Account AWS. 3. Nella sezione Account AWS, seleziona Un altro account AWS. 4. Per Account ID, inserisci l'ID account per Account B. 5. Nella pagina Aggiungi autorizzazioni, cerca e scegli la `CrossAccountAccessForMySharedDemoRepo` politica che hai creato in precedenza. 6. Scegli Next (Successivo). 7. In Role name (Nome ruolo), immettere un nome. Nota: in questo modello, il nome del ruolo IAM viene chiamato `MyCrossAccountRepositoryContributorRole` , ma puoi scegliere il nome del ruolo che preferisci. 8. Scegli Crea ruolo, quindi copia l'Amazon Resource Name (ARN) del nuovo ruolo.	AWS DevOps

Crea una policy IAM in linea nell'Account B

Operazione	Description	Competenze richieste
Allega una policy in linea al ruolo di esecuzione associato al tuo utente di SageMaker dominio nell'Account B.	1. Nel pannello di navigazione della console IAM, scegli Ruoli. 2. Cerca e scegli il ruolo di esecuzione associato all'utente del tuo dominio SageMaker AI nell'Account B. 3. Scegli Aggiungi autorizzazioni, quindi seleziona Crea policy in linea. 4. Scegli la scheda JSON. 5. Copia la seguente dichiarazione sulla politica e incollala nell'editor JSON. `{ "Version": "2012-10-17", "Statement": [ { "Sid": "VisualEditor0", "Effect": "Allow", "Action": "sts:AssumeRole", "Resource": "arn:aws:iam::<Account_A_ID>:role/<Account_A_Role_Name>" } ] }`	AWS DevOps

Operazione	Description	Competenze richieste
	6. Sostituisci `<Account_A_ID>` con l'ID dell'account A. Sostituisci `<Account_A_Role_Name>` con il nome del ruolo IAM creato in precedenza. 7. Scegliere Esamina policy. 8. In Nome, inserisci un nome per la tua politica in linea. 9. Scegli Crea policy.	

Clona il repository in SageMaker AI Studio Classic per l'account B

Operazione	Description	Competenze richieste
Crea lo script di shell in SageMaker AI Studio Classic nell'Account B.	1. Nel pannello di navigazione della SageMaker console, scegli Studio. 2. Seleziona il tuo profilo utente, quindi scegli Open Studio. 3. Nella sezione Home, scegli Open Launcher. 4. Nella sezione Utilità e file, scegli File di testo. 5. Copia lo script da SageMaker Example shell script nella sezione Informazioni aggiuntive di questo modello, quindi incolla l'istruzione nel nuovo file. Assicurati di sostituir	AWS DevOps

Operazione	Description	Competenze richieste
	e tutti i valori segnaposto nello script. 6. Fate clic con il pulsante destro del mouse sulla scheda untitled.txt del nuovo file, quindi scegliete Rinomina testo. Per Nuovo nome, immettete cross_account_git_clone.sh, quindi scegliete Rinomina.	
Richiama lo script di shell dal terminale di sistema.	1. Nella sezione Home della SageMaker console, scegli Open Launcher. 2. Nella sezione Utilità e file, scegli Terminale di sistema. 3. Nel terminale, esegui il seguente comando: `chmod u+x ./cross_account_git_clone.sh && ./cross_account_git_clone.sh` Hai clonato il tuo CodeCommit repository in un account multiplo di SageMaker AI Studio. Ora puoi eseguire tutti i comandi Git dal terminale di sistema.	AWS DevOps

Informazioni aggiuntive

Policy IAM di esempio

Se utilizzi questa politica di esempio, procedi come segue:

- Sostituisci <CodeCommit_Repository_Region> con Regione AWS for the repository.
- Sostituisci <Account_A_ID> con l'ID dell'account A.
- Sostituiscilo <CodeCommit_Repository_Name> con il nome del tuo CodeCommit repository nell'Account A.

```
{
"Version": "2012-10-17",
"Statement": [
    {
        "Effect": "Allow",
        "Action": [
            "codecommit:BatchGet*",
            "codecommit:Create*",
            "codecommit:DeleteBranch",
            "codecommit:Get*",
            "codecommit:List*",
            "codecommit:Describe*",
            "codecommit:Put*",
            "codecommit:Post*",
            "codecommit:Merge*",
            "codecommit:Test*",
            "codecommit:Update*",
            "codecommit:GitPull",
            "codecommit:GitPush"
        ],
        "Resource": [

 "arn:aws:codecommit:<CodeCommit_Repository_Region>:<Account_A_ID>:<CodeCommit_Repository_Name>
        ]
    }
]
}
```

Esempio di script di shell SageMaker AI

Se usi questo script di esempio, procedi come segue:

- Sostituisci <Account_A_ID> con l'ID dell'account A.
- Sostituiscilo <Account_A_Role_Name> con il nome del ruolo IAM che hai creato in precedenza.

- Sostituiscilo <CodeCommit_Repository_Region> con Regione AWS for the repository.
- <CodeCommit_Repository_Name>Sostituiscilo con il nome del tuo CodeCommit repository nell'Account A.

```
#!/usr/bin/env bash
#Launch from system terminal
pip install --quiet git-remote-codecommit

mkdir -p ~/.aws
touch ~/.aws/config

echo "[profile CrossAccountAccessProfile]
region = <CodeCommit_Repository_Region>
credential_source=EcsContainer
role_arn = arn:aws:iam::<Account_A_ID>:role/<Account_A_Role_Name>
output = json" > ~/.aws/config

echo '[credential "https://git-
codecommit.<CodeCommit_Repository_Region>.amazonaws.com"]
        helper = !aws codecommit credential-helper $@ --profile
 CrossAccountAccessProfile
        UseHttpPath = true' > ~/.gitconfig

git clone codecommit::<CodeCommit_Repository_Region>://
CrossAccountAccessProfile@<CodeCommit_Repository_Name>
```

Estrai automaticamente i contenuti dai file PDF utilizzando Amazon Textract

Tianxia Jia, Amazon Web Services

Riepilogo

Molte organizzazioni devono estrarre informazioni dai file PDF caricati nelle loro applicazioni aziendali. Ad esempio, un'organizzazione potrebbe aver bisogno di estrarre con precisione le informazioni dai file PDF fiscali o medici per l'analisi fiscale o l'elaborazione delle richieste mediche.

Sul cloud Amazon Web Services (AWS), Amazon Textract estrae automaticamente le informazioni (ad esempio testo stampato, moduli e tabelle) dai file PDF e produce un file in formato JSON che contiene informazioni dal file PDF originale. Puoi utilizzare Amazon Textract nella Console di gestione AWS o implementando chiamate API. Ti consigliamo di utilizzare chiamate API programmatiche per scalare ed elaborare automaticamente un gran numero di file PDF.

Quando Amazon Textract elabora un file, crea il seguente elenco di `Block` oggetti: pagine, righe e parole di testo, moduli (coppie chiave-valore), tabelle e celle ed elementi di selezione. Sono incluse anche altre informazioni sugli oggetti, ad esempio riquadri di delimitazione, intervalli di confidenza e relazioni. IDs Amazon Textract estrae le informazioni sul contenuto sotto forma di stringhe. I valori dei dati correttamente identificati e trasformati sono necessari perché possono essere utilizzati più facilmente dalle applicazioni downstream.

Questo modello descrive un step-by-step flusso di lavoro per l'utilizzo di Amazon Textract per estrarre automaticamente il contenuto dai file PDF ed elaborarlo in un output pulito. Il modello utilizza una tecnica di abbinamento dei modelli per identificare correttamente il campo, il nome chiave e le tabelle richiesti, quindi applica le correzioni post-elaborazione a ciascun tipo di dati. È possibile utilizzare questo modello per elaborare diversi tipi di file PDF e quindi ridimensionare e automatizzare questo flusso di lavoro per elaborare file PDF con un formato identico.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un bucket Amazon Simple Storage Service (Amazon S3) esistente per archiviare i file PDF dopo la conversione in formato JPEG per l'elaborazione da parte di Amazon Textract. Per ulteriori

informazioni sui bucket S3, consulta la panoramica dei bucket nella documentazione di Amazon S3.

- Il notebook `Textract_PostProcessing.ipynb` Jupyter (allegato), installato e configurato. Per ulteriori informazioni sui notebook Jupyter, consulta Creare un notebook Jupyter nella documentazione di Amazon. SageMaker
- File PDF esistenti con un formato identico.
- Una comprensione di Python.

Limitazioni

- I tuoi file PDF devono essere di buona qualità e chiaramente leggibili. Si consigliano file PDF nativi, ma è possibile utilizzare documenti scansionati convertiti in formato PDF se tutte le singole parole sono chiare. Per ulteriori informazioni su questo argomento, consulta Preelaborazione dei documenti PDF con Amazon Textract: rilevamento e rimozione di elementi visivi sul blog di AWS Machine Learning.
- Per i file multipagina, puoi utilizzare un'operazione asincrona o dividere i file PDF in un'unica pagina e utilizzare un'operazione sincrona. Per ulteriori informazioni su queste due opzioni, consulta Rilevamento e analisi del testo in documenti multipagina e Rilevamento e analisi del testo in documenti a pagina singola nella documentazione di Amazon Textract.

Architecture

Il flusso di lavoro di questo pattern esegue prima Amazon Textract su un file PDF di esempio (prima esecuzione) e poi lo esegue su file PDF con un formato identico al primo PDF (esecuzione ripetuta). Il diagramma seguente mostra il flusso di lavoro combinato First-time run e Repeat run che estrae automaticamente e ripetutamente il contenuto da file PDF con formati identici.

Il diagramma mostra il seguente flusso di lavoro per questo modello:

1. Converti un file PDF in formato JPEG e archivialo in un bucket S3.
2. Chiama l'API Amazon Textract e analizza il file JSON di risposta Amazon Textract.
3. Modifica il file JSON aggiungendo la `KeyName:DataType` coppia corretta per ogni campo obbligatorio. Crea un `TemplateJSON` file per la fase Repeat run.

4. Definite le funzioni di correzione post-elaborazione per ogni tipo di dati (ad esempio, float, integer e date).
5. Prepara i file PDF con un formato identico al tuo primo file PDF.
6. Chiama l'API Amazon Textract e analizza il codice JSON di risposta Amazon Textract.
7. Abbina il file JSON analizzato al file. `TemplateJSON`
8. Implementa le correzioni successive all'elaborazione.

Il file di output JSON finale contiene i campi corretti `KeyName` e `Value` per ogni campo obbligatorio.

Stack tecnologico Target

- Amazon SageMaker
- Simple Storage Service (Amazon S3)
- Amazon Textract

Automazione e scalabilità

Puoi automatizzare il flusso di lavoro Repeat run utilizzando una funzione AWS Lambda che avvia Amazon Textract quando viene aggiunto un nuovo file PDF ad Amazon S3. Amazon Textract esegue quindi gli script di elaborazione e l'output finale può essere salvato in una posizione di archiviazione. Per ulteriori informazioni su questo argomento, consulta Usare un trigger di Amazon S3 per richiamare una funzione Lambda nella documentazione Lambda.

Tools (Strumenti)

- Amazon SageMaker è un servizio di machine learning completamente gestito che ti aiuta a creare e addestrare modelli di machine learning in modo rapido e semplice, per poi distribuirli direttamente in un ambiente ospitato pronto per la produzione.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Textract semplifica l'aggiunta del rilevamento e dell'analisi del testo dei documenti alle tue applicazioni.

Epiche

Prima corsa

Operazione	Description	Competenze richieste
Convertire il file PDF.	Prepara il file PDF per la prima esecuzione suddividendolo in un'unica pagina e convertendolo in formato JPEG per il funzionamento sincrono di Amazon Textract (). `Syn API` **Note** Puoi anche utilizzare l'operazione asincrona di Amazon Textract (`Asyn API`) per file PDF multipagina.	Scienziato dei dati, sviluppatore
Analizza il codice JSON della risposta Amazon Textract.	Apri il notebook `Textract_PostProcessing.ipynb` Jupyter (allegato) e richiama l'API Amazon Textract utilizzando il seguente codice: `response = textract.analyze_document( Document={ 'S3Object': { 'Bucket': BUCKET, 'Name': '{}'.format(filename) } }, ...`	Scienziato dei dati, sviluppatore

Operazione	Description	Competenze richieste

	`FeatureTypes=["TABLES", "FORMS"])`
	
	Analizza la risposta JSON in un modulo e in una tabella utilizzando il codice seguente:
	
	`parseformKV=form_kv_from_JSON(response)`
	`parseformTables=get_tables_fromJSON(response)`	

Modifica il file TemplateJSON.	Modifica il codice JSON analizzato per tutte le intestazioni di tabella corrispondenti `DataType` (ad esempio, string, float, integer o date) `KeyName` e per le relative intestazioni di tabella (ad esempio e). `ColumnNames` `RowNames` Questo modello viene utilizzato per ogni singolo tipo di file PDF, il che significa che può essere riutilizzato per file PDF con un formato identico.	Scienziato dei dati, sviluppatore

Operazione	Description	Competenze richieste
Definire le funzioni di correzione e post-elaborazione.	I valori nella risposta di Amazon Textract per il `TemplateJSON` file sono stringhe. Non vi è alcuna differenziazione per data, float, numero intero o valuta. Questi valori devono essere convertiti nel tipo di dati corretto per il caso d'uso a valle. Correggi ogni tipo di dati in base al `TemplateJSON` file utilizzando il codice seguente: `finalJSON=postprocessingCorrection(parsedJSON,templateJSON)`	Scienziato dei dati, sviluppatore

Ripeti la corsa

Operazione	Description	Competenze richieste
Prepara i file PDF.	Prepara i file PDF dividendoli in un'unica pagina e convertendoli in formato JPEG per il funzionamento sincrono di Amazon Textract (). `Syn API` **Note** Puoi anche utilizzare l'operazione asincrona di Amazon Textract	Scienziato dei dati, sviluppatore

Operazione	Description	Competenze richieste
	(`Asyn API`) per file PDF multipagina.	
Chiama l'API Amazon Textract.	Chiama l'API Amazon Textract utilizzando il seguente codice: `response = textract.analyze_document( Document={ 'S3Object': { 'Bucket': BUCKET, 'Name': '{}'.format(filename) } }, FeatureTypes=["TABLES", "FORMS"])`	Scienziato dei dati, sviluppatore
Analizza il codice JSON della risposta Amazon Textract.	Analizza la risposta JSON in un modulo e in una tabella utilizzando il codice seguente: `parseformKV=form_kv_from_JSON(response) parseformTables=get_tables_fromJSON(response)`	Scienziato dei dati, sviluppatore

Operazione	Description	Competenze richieste
Carica il file TemplateJSON e abbinalo al JSON analizzato.	Utilizzate il `TemplateJSON` file per estrarre le coppie chiave-valore corrette e la tabella utilizzando i seguenti comandi: `form_kv_corrected= form_kv_correction (parseformKV,templ ateJSON) form_table_correct ed=form_Table_corr ection(parseformTa bles, templateJSON) form_kv_table_correcte d_final={**form_kv _corrected , **form_ta ble_corrected}`	Scienziato dei dati, sviluppat ore
Correzioni successive all'elabo razione.	Utilizza `DataType` le funzioni di `TemplateJSON` file e post-elaborazione per correggere i dati utilizzando il codice seguente: `finalJSON=postproc essingCorrection(f orm_kv_table_corre cted_final,templat eJSON)`	Scienziato dei dati, sviluppat ore

Risorse correlate

- Estrai automaticamente testo e dati strutturati dai documenti con Amazon Textract
- Estrai testo e dati strutturati con Amazon Textract
- Risorse Amazon Textract

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Crea un modello di previsione con avvio a freddo utilizzando DeepAR per le serie temporali in SageMaker Amazon AI Studio Lab

Ivan Cui e Eyal Shacham, Amazon Web Services

Riepilogo

Che si tratti di allocare le risorse in modo più efficiente per il traffico web, di prevedere la domanda di personale dei pazienti o di anticipare le vendite dei prodotti di un'azienda, le previsioni sono uno strumento essenziale. La previsione con avvio a freddo crea previsioni per una serie temporale con pochi dati storici, ad esempio un nuovo prodotto che è appena entrato nel mercato al dettaglio. Questo modello utilizza l'algoritmo di previsione Amazon SageMaker AI DeepAR per addestrare un modello di previsione con avviamento a freddo e dimostra come eseguire previsioni sugli articoli con avviamento a freddo.

DeepAR è un algoritmo di apprendimento supervisionato per la previsione di serie temporali scalari (unidimensionali) utilizzando reti neurali ricorrenti (RNN). DeepAR adotta l'approccio di addestrare un singolo modello congiuntamente su tutte le serie temporali delle serie temporali dei prodotti correlati.

I metodi tradizionali di previsione delle serie temporali come la media mobile integrata autoregressiva (ARIMA) o il livellamento esponenziale (ETS) si basano in larga misura sulle serie temporali storiche di ogni singolo prodotto. Pertanto, questi metodi non sono efficaci per le previsioni con avvio a freddo. Quando il set di dati contiene centinaia di serie temporali correlate, DeepAR ottiene risultati migliori rispetto ai metodi ARIMA ed ETS standard. È inoltre possibile utilizzare il modello addestrato per generare previsioni per nuove serie temporali simili a quelle su cui è stato addestrato.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS.
- Un dominio Amazon SageMaker AI.
- Un'applicazione di laboratorio Amazon SageMaker AI Studio Lab o Jupiter.
- Un bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) con autorizzazioni di lettura e scrittura.

- Conoscenza della programmazione in Python.
- Conoscenza dell'uso di un notebook Jupyter.

Limitazioni

- L'richiamo del modello previsionale senza punti dati storici restituirà un errore. L'utilizzo del modello con un numero minimo di punti dati storici restituirà previsioni imprecise con elevata affidabilità. Questo modello suggerisce un approccio per risolvere questi limiti noti delle previsioni con avvio a freddo.
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità regionale, consulta i servizi AWS per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- Python versione 3.10 o successiva.
- Il notebook del pattern è stato testato in Amazon SageMaker AI Studio su un'istanza ml.t3.medium con il kernel Python 3 (Data Science).

Architecture

Il diagramma seguente mostra i componenti del flusso di lavoro e dell'architettura per questo modello.

Il flusso di lavoro esegue le seguenti attività:

1. I file di input dei dati di addestramento e test vengono sintetizzati e quindi caricati in un bucket Amazon S3. Questi dati includono più serie temporali con caratteristiche categoriche e dinamiche, insieme a valori target (da prevedere). Il notebook Jupyter visualizza i dati per comprendere meglio i requisiti dei dati di allenamento e i valori previsti previsti.
2. Viene creato un job di ottimizzazione degli iperparametri per addestrare il modello e trovare il modello migliore in base a metriche predefinite.
3. I file di input vengono scaricati dal bucket Amazon S3 in ogni istanza dei processi di ottimizzazione degli iperparametri.
4. Dopo che il tuner job ha selezionato il modello migliore in base alla soglia predefinita del tuner, il modello viene distribuito come endpoint AI. SageMaker

5. Il modello implementato è quindi pronto per essere richiamato dove le sue previsioni vengono convalidate rispetto ai dati di test.

Il notebook dimostra quanto bene il modello predice i valori target quando è disponibile un numero adeguato di punti dati storici. Tuttavia, quando richiamiamo il modello con un minor numero di punti dati storici (che rappresentano un prodotto freddo), le previsioni del modello non corrispondono ai dati di test originali nemmeno entro i livelli di confidenza del modello. Nello schema, viene creato un nuovo modello per prodotti freddi in cui la lunghezza del contesto iniziale (punti previsti) è definita come la quantità di punti storici disponibili e un nuovo modello viene addestrato iterativamente man mano che vengono acquisiti nuovi punti dati. Il taccuino mostra che il modello avrà previsioni accurate purché la quantità di punti dati storici sia vicina alla lunghezza del contesto.

Tools (Strumenti)

Servizi AWS

- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon SageMaker AI è un servizio di machine learning (ML) gestito che ti aiuta a creare e addestrare modelli di machine learning per poi distribuirli in un ambiente ospitato pronto per la produzione.
- Amazon SageMaker AI Studio è un ambiente di sviluppo integrato (IDE) basato sul Web per il machine learning che ti consente di creare, addestrare, eseguire il debug, distribuire e monitorare i tuoi modelli di machine learning.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri strumenti

- Python è un linguaggio di programmazione per computer generico.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub DeepAr- ColdProduct -Pattern.

Best practice

- Addestra il tuo modello in un ambiente virtuale e usa sempre il controllo della versione per il massimo sforzo di riproducibilità.
- Includi quante più funzionalità categoriali di alta qualità possibile per ottenere il modello predittivo più elevato.
- Assicurati che i metadati contengano elementi categoriali simili in modo che il modello possa dedurre in modo adeguato le previsioni dei prodotti con avvio a freddo.
- Esegui un lavoro di ottimizzazione degli iperparametri per ottenere il modello predittivo più elevato.
- In questo modello, il modello creato ha una durata del contesto di 24 ore, il che significa che prevede le 24 ore successive. Se provi a prevedere le prossime 24 ore quando storicamente disponi di meno di 24 ore di dati, la precisione di previsione del modello diminuisce linearmente in base alla quantità di punti dati storici. Per mitigare questo problema, crea un nuovo modello per ogni set di dati storici fino a raggiungere la lunghezza di previsione (contesto) desiderata. Ad esempio, iniziate con un modello di lunghezza contestuale di 2 ore, quindi aumentate progressivamente il modello fino a 4 ore, 8 ore, 16 ore e 24 ore.

Epiche

Avvia la tua SageMaker applicazione AI Studio Classic

Operazione	Description	Competenze richieste
Avvia il tuo ambiente notebook.	1. Accedi a e apri la home page di SageMaker AI Studio. Console di gestione AWS Quindi scegli Open Studio. 2. Nel riquadro di navigazio ne a sinistra, scegli l'icona Studio Classic in Applicazi oni. Quindi, scegli il pulsante Apri nell'elenco delle applicazioni.	Data Scientist

Operazione	Description	Competenze richieste
	Per ulteriori informazioni, consulta Launch Amazon SageMaker AI Studio nella documentazione sull' SageMaker intelligenza artificiale.	

Crea e attiva il notebook

Operazione	Description	Competenze richieste
Configura il tuo ambiente virtuale per la formazione dei modelli.	Per configurare l'ambiente virtuale per l'addestramento dei modelli, procedi come segue: 1. Scarica il `deepar_synthetic.ipynb` notebook dal GitHub repository di questo pattern sul tuo computer locale. 2. In Amazon SageMaker AI Studio Classic, scegli l'icona Carica file dalla barra dei menu di Studio Classic e seleziona il notebook scaricato. 3. Scegli il taccuino nel File Browser nel riquadro di navigazione a sinistra. Segui le istruzioni per configurare un ambiente notebook. Seleziona	Data Scientist

Operazione	Description	Competenze richieste
	l'immagine Data Science 3.0 e il kernel Python 3. Per ulteriori informazioni, consulta Caricare file su SageMaker AI Studio Classic nella documentazione SageMaker AI.	
Crea e convalida un modello previsionale.	• Segui le istruzioni riportate nel taccuino per creare i dati di addestramento e test, addestrare il modello e quindi richiamarlo. • Osserva quanto sono accurate le previsioni del modello quando viene fornito con dati storici adeguati.	Data Scientist

Risorse correlate

- Iperparametri DeepAR
- Previsione della domanda di introduzione di nuovi prodotti utilizzando servizi di apprendimento automatico AWS
- Avvia Amazon SageMaker AI Studio Classic
- Usa l'algoritmo di SageMaker previsione AI DeepAR

Crea un MLOps flusso di lavoro usando Amazon SageMaker AI e Azure DevOps

Deepika Kumar, Sara van de Moosdijk e Philips Kokoh Prasetyo, Amazon Web Services

Riepilogo

Le operazioni di apprendimento automatico (MLOps) sono un insieme di pratiche che automatizzano e semplificano i flussi di lavoro e le implementazioni di machine learning (ML). MLOps si concentra sull'automazione del ciclo di vita del machine learning. Aiuta a garantire che i modelli non vengano solo sviluppati, ma anche implementati, monitorati e riqualificati in modo sistematico e ripetuto. Apporta principi al machine learning DevOps . MLOps si traduce in un'implementazione più rapida dei modelli di machine learning, una migliore precisione nel tempo e una maggiore garanzia che offrano un reale valore aziendale.

Le organizzazioni spesso dispongono di DevOps strumenti e soluzioni di archiviazione dei dati già esistenti prima di iniziare il loro MLOps percorso. Questo modello mostra come sfruttare i punti di forza di Microsoft Azure e. AWS Ti aiuta a integrare Azure DevOps con Amazon SageMaker AI per creare un MLOps flusso di lavoro.

La soluzione semplifica il lavoro tra Azure e. AWS Puoi usare Azure per lo sviluppo e AWS per l'apprendimento automatico. Promuove un processo efficace per creare modelli di apprendimento automatico dall'inizio alla fine, tra cui la gestione dei dati, la formazione e la distribuzione. AWS Per garantire l'efficienza, gestisci questi processi tramite Azure DevOps pipelines. La soluzione è applicabile alle operazioni dei modelli di base (FMOps) e alle operazioni con modelli linguistici di grandi dimensioni (LLMOps) nell'intelligenza artificiale generativa, che include ottimizzazione fine, database vettoriali e gestione dei prompt.

Prerequisiti e limitazioni

Prerequisiti

- Sottoscrizione ad Azure: accesso ai servizi di Azure, come Azure DevOps, per configurare le pipeline di integrazione e distribuzione continua (CI/CD).
- Account AWS attivo: autorizzazioni all'uso di ciò che Servizi AWS viene utilizzato in questo modello.
- Dati: accesso ai dati storici per addestrare il modello di apprendimento automatico.

- Familiarità con i concetti di machine learning: comprensione di Python, Jupyter Notebooks e sviluppo di modelli di machine learning.
- Configurazione di sicurezza: configurazione corretta di ruoli, policy e autorizzazioni su Azure e AWS per garantire il trasferimento e l'accesso sicuri ai dati.
- (Facoltativo) Database vettoriale: se utilizzi un approccio Retrieval Augmented Generation (RAG) e un servizio di terze parti per il database vettoriale, devi accedere al database vettoriale esterno.

Limitazioni

- Questa guida non tratta i trasferimenti sicuri di dati tra cloud. Per ulteriori informazioni sui trasferimenti di dati tra cloud, consulta AWS Soluzioni per ambienti ibridi e multicloud.
- Le soluzioni multicloud possono aumentare la latenza per l'elaborazione dei dati in tempo reale e l'inferenza dei modelli.
- Questa guida fornisce un esempio di architettura multi-account. MLOps Sono necessari aggiustamenti in base all'apprendimento automatico e AWS alla strategia.
- Questa guida non descrive l'uso di AI/ML servizi diversi da Amazon SageMaker AI.
- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Architecture

Architettura di Target

L'architettura di destinazione integra Azure con DevOps Amazon SageMaker AI, creando un flusso di lavoro ML cross-cloud. Usa Azure per i CI/CD processi e l' SageMaker intelligenza artificiale per la formazione e la distribuzione dei modelli ML. Descrive il processo di acquisizione dei dati (da fonti come Amazon S3, Snowflake e Azure Data Lake) attraverso la creazione e la distribuzione di modelli. I componenti chiave includono CI/CD pipeline per la creazione e l'implementazione di modelli, la preparazione dei dati, la gestione dell'infrastruttura e Amazon SageMaker AI per la formazione e l'ottimizzazione, la valutazione e l'implementazione di modelli ML. Questa architettura è progettata per fornire flussi di lavoro ML efficienti, automatizzati e scalabili su piattaforme cloud.

L'architettura è composta dai seguenti componenti:

1. I data scientist eseguono esperimenti di machine learning nell'account di sviluppo per esplorare diversi approcci ai casi d'uso del machine learning utilizzando varie fonti di dati. I data scientist eseguono test unitari e prove e, per tenere traccia dei loro esperimenti, possono utilizzare Amazon SageMaker AI con MLflow. Nello sviluppo di modelli di intelligenza artificiale generativa, i data scientist perfezionano i modelli di base dell'Amazon SageMaker AI JumpStart Model Hub. Dopo la valutazione del modello, i data scientist inviano e uniscono il codice nel repository Model Build, ospitato in Azure. DevOps Questo repository contiene il codice per una pipeline di creazione di modelli in più fasi.
2. In Azure DevOps, la pipeline Model Build, che fornisce l'integrazione continua (CI), può essere attivata automaticamente o manualmente dopo l'unione del codice nel ramo principale. Nell'account Automation, questa operazione attiva la pipeline di SageMaker intelligenza artificiale per la preelaborazione dei dati, l'addestramento e la messa a punto dei modelli, la valutazione del modello e la registrazione condizionale del modello in base alla precisione.
3. L'account Automation è un account centrale tra le piattaforme ML che ospita ambienti ML (Amazon ECR), modelli (Amazon S3), metadati dei modelli (AI Model Registry), funzionalità SageMaker SageMaker (AI Feature Store), pipeline automatizzate (AI Pipelines) e ML log insights SageMaker (). CloudWatch Per un carico di lavoro di intelligenza artificiale generativo, potresti aver bisogno di valutazioni aggiuntive per i prompt nelle applicazioni downstream. Un'applicazione di gestione tempestiva aiuta a semplificare e automatizzare il processo. Questo account consente la riutilizzabilità delle risorse ML e applica le migliori pratiche per accelerare la distribuzione dei casi d'uso del machine learning.
4. L'ultima versione del modello viene aggiunta all' SageMaker AI Model Registry per la revisione. Tiene traccia delle versioni del modello e dei rispettivi artefatti (lignaggio e metadati). Gestisce inoltre lo stato del modello (approvazione, rifiuto o in sospeso) e gestisce la versione per la distribuzione a valle.
5. Dopo l'approvazione di un modello addestrato in Model Registry tramite l'interfaccia di studio o una chiamata API, è possibile inviare un evento ad Amazon. EventBridge EventBridge avvia la pipeline Model Deploy su Azure. DevOps
6. La pipeline Model Deploy, che fornisce la distribuzione continua (CD), controlla l'origine dal repository Model Deploy. Il codice sorgente contiene il codice, la configurazione per la distribuzione del modello e gli script di test per i benchmark di qualità. La pipeline Model Deploy può essere personalizzata in base al tipo di inferenza.
7. Dopo i controlli di qualità, la pipeline Model Deploy distribuisce il modello nell'account Staging. L'account Staging è una copia dell'account Production e viene utilizzato per i test e la valutazione dell'integrazione. Per una trasformazione in batch, la pipeline Model Deploy può aggiornare

automaticamente il processo di inferenza in batch per utilizzare l'ultima versione del modello approvata. Per un'inferenza in tempo reale, senza server o asincrona, configura o aggiorna il rispettivo endpoint del modello.

8. Dopo aver eseguito con successo il test nell'account Staging, un modello può essere distribuito nell'account di produzione mediante l'approvazione manuale tramite la pipeline Model Deploy. Questa pipeline fornisce un endpoint di produzione nella fase di implementazione verso la produzione, incluso il monitoraggio del modello e un meccanismo di feedback dei dati.
9. Dopo che il modello è in produzione, utilizza strumenti come SageMaker AI Model Monitor e SageMaker AI Clarify per identificare distorsioni, rilevare deviazioni e monitorare continuamente le prestazioni del modello.

Automazione e scalabilità

Utilizza l'infrastruttura come codice (IaC) per l'implementazione automatica su più account e ambienti. Automatizzando il processo di configurazione di un MLOps flusso di lavoro, è possibile separare gli ambienti utilizzati dai team di machine learning che lavorano su progetti diversi. AWS CloudFormationti aiuta a modellare, fornire e gestire AWS le risorse trattando l'infrastruttura come codice.

Tools (Strumenti)

Servizi AWS

- Amazon SageMaker AI è un servizio di machine learning gestito che ti aiuta a creare e addestrare modelli di machine learning per poi distribuirli in un ambiente ospitato pronto per la produzione.
- AWS Glueè un servizio di estrazione, trasformazione e caricamento (ETL) completamente gestito. Ti aiuta a classificare, pulire, arricchire e spostare i dati in modo affidabile tra archivi di dati e flussi di dati.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati. In questo modello, Amazon S3 viene utilizzato per l'archiviazione dei dati e integrato con l' SageMaker intelligenza artificiale per l'addestramento dei modelli e gli oggetti del modello.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi. In questo modello, Lambda viene utilizzata per le attività di pre-elaborazione e post-elaborazione dei dati.

- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile. In questo modello, archivia i contenitori Docker che l' SageMaker intelligenza artificiale utilizza come ambienti di formazione e distribuzione.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. In questo modello, EventBridge orchestra flussi di lavoro basati sugli eventi o basati sul tempo che avviano la riqualificazione o la distribuzione automatiche del modello.
- Amazon API Gateway ti aiuta a creare, pubblicare, gestire, monitorare e proteggere REST, HTTP e WebSocket APIs su qualsiasi scala. In questo modello, viene utilizzato per creare un unico punto di ingresso rivolto verso l'esterno per SageMaker gli endpoint AI.
- Per le applicazioni RAG, puoi utilizzare Servizi AWS, come Amazon OpenSearch Service e Amazon RDS for PostgreSQL, per archiviare gli incorporamenti vettoriali che forniscono all'LLM i tuoi dati interni.

Altri strumenti

- Azure ti DevOps aiuta a gestire le CI/CD pipeline e a facilitare la compilazione, i test e la distribuzione di codice.
- Azure Data Lake Storage o Snowflake sono possibili fonti di dati di formazione di terze parti per i modelli di machine learning.
- Pinecone, Milvus o ChromaDB sono possibili database vettoriali di terze parti per archiviare incorporamenti vettoriali.

Best practice

Prima di implementare qualsiasi componente di questo flusso di lavoro multicloud, completa le seguenti attività: MLOps

- Definisci e comprendi il flusso di lavoro di machine learning e gli strumenti necessari per supportarlo. Casi d'uso diversi richiedono flussi di lavoro e componenti diversi. Ad esempio, un feature store potrebbe essere necessario per il riutilizzo delle funzionalità e l'inferenza a bassa latenza in un caso d'uso di personalizzazione, ma potrebbe non essere necessario per altri casi d'uso. Per personalizzare con successo l'architettura è necessario comprendere il flusso di lavoro di destinazione, i requisiti dei casi d'uso e i metodi di collaborazione preferiti dal team di data science.

- Crea una chiara separazione delle responsabilità per ogni componente dell'architettura. La diffusione dello storage dei dati tra Azure Data Lake Storage, Snowflake e Amazon S3 può aumentare la complessità e i costi. Se possibile, scegli un meccanismo di archiviazione coerente. Allo stesso modo, evita di usare una combinazione di DevOps servizi Azure e AWS o una combinazione di servizi Azure e AWS ML.
- Scegli uno o più modelli e set di dati esistenti per eseguire end-to-end test del flusso di lavoro. MLOps Gli artefatti dei test dovrebbero riflettere casi d'uso reali che i team di data science sviluppano quando la piattaforma entra in produzione.

Epiche

Progetta la tua MLOps architettura

Operazione	Description	Competenze richieste
Identifica le fonti di dati.	In base ai casi d'uso attuali e futuri, alle fonti di dati disponibili e ai tipi di dati (come i dati riservati), documenta le fonti di dati che devono essere integrate con la MLOps piattaforma. I dati possono essere archiviati in Amazon S3, Azure Data Lake Storage, Snowflake o altre fonti. Per i carichi di lavoro di intelligenza artificiale generativa, i dati potrebbero includere anche una base di conoscenze su cui si basa la risposta generata. Questi dati vengono archiviati come incorporamenti vettoriali nei database vettoriali. Crea un piano per integrare queste fonti con la tua piattaforma e	Ingegnere dei dati, scienziato dei dati, architetto del cloud

Operazione	Description	Competenze richieste
	garantire l'accesso alle risorse corrette.	
Scegli i servizi applicabili.	Personalizza l'architettura aggiungendo o rimuovendo servizi in base al flusso di lavoro desiderato del team di data science, alle fonti di dati applicabili e all'architettura cloud esistente. Ad esempio, i data engineer e i data scientist possono eseguire la preelaborazione dei dati e l'ingegneria delle funzionalità in SageMaker AI o Amazon EMR. AWS GlueÈ improbabile che siano necessari tutti e tre i servizi.	Amministratore AWS, ingegnere dei dati, scienziato dei dati, ingegnere ML

Operazione	Description	Competenze richieste
Analizza i requisiti di sicurezza .	Raccogli e documenta i requisiti di sicurezza. Ciò include la determinazione di: • Quali team o ingegneri possono accedere a fonti di dati specifiche • Quali team o ingegneri possono accedere a modelli di base preformati • Se i team sono autorizzati ad accedere al codice e ai modelli di altri team • Quali autorizzazioni (se ce ne sono) devono avere i membri del team per gli account non in fase di sviluppo • Quali misure di sicurezza devono essere implement ate per il trasferimento di dati tra cloud Per ulteriori informazioni sulla protezione dei carichi di lavoro di intelligenza artificiale generativa, consulta Protegger e l'IA generativa: un'introd uzione alla Generative AI Security Scoping Matrix (post sul blog).AWS	Amministratore AWS, architett o cloud

Configurare AWS Organizations

Operazione	Description	Competenze richieste
Configurare AWS Organizations.	Configura AWS Organizations sulla radice Account AWS. Questo ti aiuta a gestire gli account successivi che crei come parte di una MLOps strategia multi-account. Per ulteriori informazioni, consulta la documentazione relativa ad AWS Organizations.	Amministratore AWS

Configura l'ambiente di sviluppo e il controllo delle versioni

Operazione	Description	Competenze richieste
Crea un account di AWS sviluppo.	Crea un ambiente Account AWS in cui i data engineer e i data scientist abbiano le autorizzazioni per speriment are e creare modelli ML. Per istruzioni, consulta Creazione di un account membro nell'organizzazione nella AWS Organizations documenta zione.	Amministratore AWS
Crea un repository Model Build.	Crea un repository Git in Azure in cui i data scientist possono inviare il codice di compilazione e distribuz ione del modello una volta completata la fase di sperimentazione. Per istruzion	DevOps ingegnere, ingegnere ML

Operazione	Description	Competenze richieste
	i, vedi Configurare un repository Git nella documentazione di Azure. DevOps	
Crea un repository Model Deploy.	Crea un repository Git in Azure in cui archiviare codice e modelli di distribuzione standard. Dovrebbe includere il codice per ogni opzione di distribuzione utilizzata dall'organizzazione, come identificato nella fase di progettazione. Ad esempio, dovrebbe includere endpoint in tempo reale, endpoint asincroni, inferenza senza server o trasformazioni in batch. Per istruzioni, vedi Configurare un repository Git nella documentazione di Azure. DevOps	DevOps ingegnere, ingegnere ML
Crea un repository Amazon ECR.	Configura un repository Amazon ECR che archivia gli ambienti ML approvati come immagini Docker. Consenti ai data scientist e agli ingegneri ML di definire nuovi ambienti. Per istruzioni, consulta Creazione di un repository privato nella documentazione di Amazon ECR.	Ingegnere ML

Operazione	Description	Competenze richieste
Configura SageMaker AI Studio.	Configura SageMaker AI Studio sull'account di sviluppo in base ai requisiti di sicurezza precedentemente definiti, agli strumenti di data science preferiti (come MLflow) e all'ambiente di sviluppo integrato (IDE) preferito. Utilizza le configurazioni del ciclo di vita per automatizzare l'installazione delle funzionalità chiave e creare un ambiente di sviluppo uniforme per i data scientist. Per ulteriori informazioni, consulta Amazon SageMaker AI Studio e server di MLflow tracciamento nella documentazione SageMaker AI.	Scienziato dei dati, ingegnere ML, ingegnere tempestivo

Integra le pipeline CI/CD

Operazione	Description	Competenze richieste
Crea un account Automation.	Crea un Account AWS luogo in cui vengono eseguite pipeline e lavori automatiz zati. Puoi concedere ai team di data science l'accesso in lettura a questo account. Per istruzioni, vedi Creazione di un account membro nella tua organizzazione nella AWS	Amministratore AWS

Operazione	Description	Competenze richieste
	Organizations documenta zione.	
Configura un registro dei modelli.	Configura SageMaker AI Model Registry nell'account Automation. Questo registro memorizza i metadati per i modelli ML e aiuta determinati data scientist o responsabili di team ad approvare o rifiutare i modelli. Per ulteriori informazi oni, consulta Registrare e distribuire modelli con Model Registry nella documentazione AI. SageMaker	Ingegnere ML
Crea una pipeline di Model Build.	Crea una CI/CD pipeline in Azure che si avvia manualmen te o automaticamente quando il codice viene inviato al repository Model Build. La pipeline dovrebbe controlla re il codice sorgente e creare o aggiornare una pipeline SageMaker AI nell'acco unt Automation. La pipeline dovrebbe aggiungere un nuovo modello al registro dei modelli. Per altre informazioni sulla creazione di una pipeline, consulta la documentazione di Azure Pipelines.	DevOps ingegnere, ingegnere ML

Crea lo stack di implementazione

Operazione	Description	Competenze richieste
Crea account AWS di gestione temporanea e di distribuzione.	Crea Account AWS per l'allestimento e la distribuzione di modelli ML. Questi account devono essere identici per consentire un test accurato dei modelli in fase di staging prima di passare alla produzione. Puoi concedere ai team di data science l'accesso in lettura all'account di staging. Per istruzioni, consulta Creazione di un account membro nella tua organizzazione nella AWS Organizations documentazione.	Amministratore AWS
Configura i bucket S3 per il monitoraggio dei modelli.	Completa questo passaggio se desideri abilitare il monitoraggio dei modelli distribuiti creati dalla pipeline Model Deploy. Crea bucket Amazon S3 per archiviare i dati di input e output. Per ulteriori informazioni sulla creazione di bucket S3, consulta Creazione di un bucket nella documentazione di Amazon S3. Imposta le autorizzazioni per più account in modo che i processi di monitoraggio automatico del modello vengano eseguiti	Ingegnere ML

Operazione	Description	Competenze richieste
	nell'account Automation. Per ulteriori informazioni, consulta Monitoraggio della qualità dei dati e dei modelli nella documentazione SageMaker AI.	
Crea una pipeline Model Deploy.	Crea una CI/CD pipeline in Azure che inizia quando un modello viene approvato nel registro dei modelli. La pipeline dovrebbe verificare il codice sorgente e gli elementi del modello, creare i modelli di infrastruttura per la distribuzione del modello negli account di gestione temporanea e di produzione, distribuire il modello nell'account di gestione temporanea, eseguire test automatici, attendere l'approvazione manuale e distribuire il modello approvato nell'account di produzione. Per altre informazioni sulla creazione di una pipeline, consulta la documentazione di Azure Pipelines.	DevOps ingegnere, ingegnere ML

(Facoltativo) Automatizza l'infrastruttura dell'ambiente ML

Operazione	Description	Competenze richieste
Compilazione AWS CDK o CloudFormation modelli.	Definisci AWS Cloud Development Kit (AWS CDK) i AWS CloudFormation nostri modelli per tutti gli ambienti che devono essere implement ati automaticamente. Ciò potrebbe includere l'ambient e di sviluppo, l'ambiente di automazione e gli ambienti di gestione e distribuzione. Per ulteriori informazioni, consulta la CloudFormationdocumenta zione AWS CDKe.	AWS DevOps
Crea una pipeline di infrastru ttura.	Crea una CI/CD pipeline in Azure per la distribuz ione dell'infrastruttura. Un amministratore può avviare questa pipeline per creare nuovi ambienti Account AWS e configurare gli ambienti richiesti dal team ML.	DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
Monitoraggio e rilevamento della deriva insufficienti: un monitoraggio inadeguato può comportare il mancato rilevamento dei problemi di prestazioni del modello o la deriva dei dati.	Rafforza i framework di monitoraggio con strumenti come Amazon CloudWatch, SageMaker AI Model Monitor e SageMaker AI Clarify. Configura gli avvisi per un'azione immediata sui problemi identificati.

Problema	Soluzione
Errori di attivazione della pipeline CI: la pipeline CI in Azure DevOps potrebbe non essere attivata al momento dell'unione del codice a causa di una configurazione errata.	Controlla le impostazioni del DevOps progetto Azure per assicurarti che i webhook siano configurati correttamente e puntino agli endpoint AI corretti. SageMaker
Governance: l'account di automazione centrale potrebbe non applicare le migliori pratiche su tutte le piattaforme ML, con conseguenti flussi di lavoro incoerenti.	Controlla le impostazioni dell'account Automatio n, assicurandoti che tutti gli ambienti e i modelli ML siano conformi alle migliori pratiche e politiche predefinite.
Ritardi nell'approvazione del registro dei modelli: ciò si verifica quando si verifica un ritardo nella verifica e nell'approvazione del modello, perché le persone impiegano del tempo per esaminarlo o a causa di problemi tecnici.	Implementa un sistema di notifica per avvisare le parti interessate dei modelli in attesa di approvazione e semplifica il processo di revisione.
Errori degli eventi di implementazione del modello: gli eventi inviati per avviare le pipeline di distribuzione del modello potrebbero non riuscire, causando ritardi nella distribuzione.	Verifica che Amazon EventBridge disponga delle autorizzazioni e dei modelli di eventi corretti per richiamare con successo le pipeline di Azure DevOps .
Colli di bottiglia nell'implementazione della produzione: i processi di approvazione manuali possono creare colli di bottiglia, ritardando l'implementazione dei modelli in produzione.	Ottimizza il flusso di lavoro di approvazione all'interno della pipeline di implementazione del modello, promuovendo revisioni tempestive e canali di comunicazione chiari.

Risorse correlate

AWS documentazione

- Documentazione Amazon SageMaker AI
- Machine Learning Lens (AWS Well Architected Framework)
- Pianificazione per il successo MLOps (guida AWS prescrittiva)

Altre risorse AWS

- MLOps roadmap di base per le aziende con Amazon SageMaker AI (post AWS sul blog)
- AWS Summit ANZ 2022 - End-to-end MLOps per architetti (video) YouTube
- FMOps/LLMOps: Rendi operativa l'IA generativa e le differenze con MLOps (post sul blog)AWS
- Rendi operativa la valutazione LLM su larga scala utilizzando Amazon SageMaker AI Clarify e MLOps i servizi (post sul blog)AWS
- Il ruolo dei database vettoriali nelle applicazioni di intelligenza artificiale generativa (post sul blog)AWS

Documentazione di Azure

- Documentazione di Azure DevOps
- Documentazione di Azure Pipelines

Configura la registrazione delle chiamate dei modelli in Amazon Bedrock utilizzando AWS CloudFormation

Vikramaditya Bhatnagar, Amazon Web Services

Riepilogo

Puoi configurare Amazon Bedrock per raccogliere registri delle chiamate, dati di input del modello e dati di output del modello per tutte le chiamate del modello nel tuo. Account AWS Si tratta di una best practice per creare solide applicazioni di intelligenza artificiale generativa con Amazon Bedrock. Puoi archiviare i log delle chiamate dei modelli in un gruppo di log Amazon CloudWatch Logs, in un bucket Amazon Simple Storage Service (Amazon S3) o in entrambi. La presenza dei dati di log in CloudWatch Logs ti aiuta a creare filtri metrici, allarmi e dashboard personalizzati. Amazon S3 è ideale per la replica dei dati tra Regioni AWS o per lo storage a lungo termine, in base alle politiche dell'organizzazione.

Questo modello fornisce un AWS CloudFormation modello di esempio che utilizza un approccio Infrastructure as Code (IaC) per configurare la registrazione delle invocazioni dei modelli per Amazon Bedrock. Il modello configura l'archiviazione dei log sia in CloudWatch Logs che in Amazon S3.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Le seguenti autorizzazioni:
  - Autorizzazioni per creare pile CloudFormation
  - Autorizzazioni per accedere ad Amazon Bedrock
  - Autorizzazioni per creare e accedere ai bucket Amazon S3
  - Autorizzazioni per creare e accedere ai gruppi di log di Logs CloudWatch
  - Autorizzazioni per creare e accedere alle funzioni AWS Lambda
  - Autorizzazioni per creare e accedere a AWS Key Management Service () chiavi AWS KMS

Limitazioni

Questo modello registra le chiamate del modello sia su CloudWatch Logs che su Amazon S3. Non supporta la scelta di uno solo di questi due servizi.

Architecture

Architettura Target

Il CloudFormation modello fornisce le seguenti risorse nel tuo target Account AWS:

- Un gruppo di CloudWatch log Logs per l'archiviazione dei log di invocazione del modello
- Un bucket Amazon S3 per l'archiviazione dei log di invocazione dei modelli e una policy del bucket corrispondente
- Un bucket Amazon S3 per l'archiviazione dei log di accesso lato server e una politica del bucket corrispondente
- Una AWS Lambda funzione che configura le impostazioni di registrazione in Amazon Bedrock
- Un alias chiave AWS KMS key e un alias corrispondente
- Un ruolo di servizio AWS Identity and Access Management (IAM) per Amazon Bedrock

Il diagramma seguente mostra come vengono archiviati i log delle chiamate dopo aver distribuito lo stack associato a questo pattern. CloudFormation Amazon Bedrock pubblica dati di log quando il modello di base fornisce testo, un'immagine, un video o dati di incorporamento. Come illustrato nel diagramma, i bucket Amazon S3 e CloudWatch il gruppo di log Logs sono crittografati con un. AWS KMS key

Il diagramma mostra il flusso di lavoro seguente:

1. Un utente invia una query a un modello di base in Amazon Bedrock.
2. Amazon Bedrock assume il ruolo di servizio IAM.
3. Amazon Bedrock genera dati di log e li archivia in un gruppo di log CloudWatch Logs e in un bucket Amazon S3.
4. Se un utente legge, carica o elimina qualsiasi file nel bucket Amazon S3 che contiene i log di invocazione del modello, tali attività vengono registrate in un altro bucket Amazon S3 per i log di accesso lato server.

Automazione e scalabilità

Per scalare questa soluzione, puoi distribuire il CloudFormation modello come uno stack impostato su più Regioni AWS e. Account AWS Per ulteriori informazioni, consulta Gestire gli stack tra account e regioni StackSets nella documentazione. CloudFormation

Tools (Strumenti)

Servizi AWS

- Amazon Bedrock è un servizio completamente gestito che rende disponibili per l'uso modelli di base ad alte prestazioni (FMs) delle principali aziende di intelligenza artificiale e di Amazon tramite un'API unificata.
- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle per tutto il loro ciclo di vita tra e. Account AWS Regioni AWS
- Amazon CloudWatch Logs ti aiuta a centralizzare i log di tutti i tuoi sistemi e applicazioni, Servizi AWS così puoi monitorarli e archiviarli in modo sicuro.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è un servizio di storage di oggetti che offre scalabilità, disponibilità dei dati, sicurezza e prestazioni all'avanguardia nel settore.

Altri strumenti

- Git è un sistema di controllo delle versioni distribuito e open source.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub enable-bedrock-logging-using-cloudformation.

Epiche

Crea lo stack CloudFormation

Operazione	Description	Competenze richieste
Scarica il CloudFormation modello.	Scarica il CloudFormation modello dal GitHub repository.	Architetto del cloud
Implementa il modello.	Crea uno stack nell'account e nella regione di destinazione. Nella sezione Parametri, specifica i valori per i parametri definiti nel modello. Per istruzioni, consultate Creazione di uno stack nella CloudFormation documentazione.	Architetto del cloud

Test della soluzione

Operazione	Description	Competenze richieste
Abilita l'accesso al modello.	In Amazon Bedrock, aggiungi l'accesso al modello di base. Per istruzioni, consulta Aggiungere o rimuovere l'accesso ai modelli Amazon Bedrock Foundation nella documentazione di Amazon Bedrock.	Architetto del cloud
Esegui un prompt di esempio.	Nei playgrounds di Amazon Bedrock, esegui un prompt di esempio. Per istruzioni, consulta Generare risposte nella console utilizzan	Architetto del cloud

Operazione	Description	Competenze richieste
	do campi da gioco nella documentazione di Amazon Bedrock.	
Rivedi la configurazione di registrazione.	1. Accedi alla console Amazon Bedrock. 2. Nella barra di navigazione, scegli Regione AWS dove hai distribuito lo CloudFormation stack. 3. Nel riquadro di navigazione a sinistra, in Configurazioni Bedrock, scegli Impostazioni. 4. Conferma quanto segue: • La registrazione delle chiamate del modello è abilitata. • Tutti i tipi di dati sono selezionati. • Per le destinazioni di registrazione, sono selezionati sia S3 che CloudWatch Logs.	Architetto del cloud
Esamina il bucket Amazon S3.	1. Nella sezione Configurazioni S3, scegli Browse S3. Questo apre il bucket di destinazione nella console Amazon S3. 2. Verifica che siano presenti i dati di registrazione per il prompt di esempio eseguito in precedenza.	Architetto del cloud

Operazione	Description	Competenze richieste
Esamina il gruppo di log.	1. Torna alla pagina Impostazioni nella console Amazon Bedrock. 2. Nella sezione Configurazioni CloudWatch dei log, rivedi le impostazioni per il gruppo CloudWatch Logs log. Prendi nota del nome del gruppo di log. 3. Apri la CloudWatch console. 4. Nel riquadro di navigazione, in Log, scegli Gruppi di log. 5. Scegli il nome del gruppo di log in cui Amazon Bedrock pubblica i dati di log. 6. Verifica che i dati di registrazione siano presenti per il prompt di esempio eseguito in precedenza.	Architetto del cloud

Risorse correlate

AWS documentazione

- Accesso a un bucket Amazon S3 (documentazione Amazon S3)
- Creazione e gestione degli stack (documentazione) CloudFormation
- Richiamata del modello di monitoraggio (documentazione Amazon Bedrock)
- Utilizzo di gruppi di log e flussi di log (documentazione relativa ai log) CloudWatch

AWS post di blog

- Monitoraggio di applicazioni di intelligenza artificiale generativa utilizzando Amazon Bedrock e l'integrazione con Amazon CloudWatch
- Best practice per creare solide applicazioni di intelligenza artificiale generativa con Amazon Bedrock Agents — Parte 1
- Best practice per creare solide applicazioni di intelligenza artificiale generativa con Amazon Bedrock Agents — Parte 2

Crea un'immagine di contenitore Docker personalizzata SageMaker e usala per l'addestramento dei modelli in AWS Step Functions

Julia Bluszcz, Aubrey Oosthuizen, Mohan Gowda Purushothama, Neha Sharma e Mateusz Zaremba, Amazon Web Services

Riepilogo

Questo modello mostra come creare un'immagine di contenitore Docker per Amazon SageMaker e utilizzarla per un modello di formazione in AWS Step Functions. Impacchettando algoritmi personalizzati in un contenitore, puoi eseguire quasi tutto il codice nell' SageMaker ambiente, indipendentemente dal linguaggio di programmazione, dal framework o dalle dipendenze.

Nel SageMaker notebook di esempio fornito, l'immagine del contenitore Docker personalizzato viene archiviata in Amazon Elastic Container Registry (Amazon ECR). Step Functions utilizza quindi il contenitore archiviato in Amazon ECR per eseguire uno script di elaborazione Python per. SageMaker Quindi, il contenitore esporta il modello in Amazon Simple Storage Service (Amazon S3).

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un ruolo AWS Identity and Access Management (IAM) per chi utilizza SageMaker le autorizzazioni di Amazon S3
- Un ruolo IAM per Step Functions
- Familiarità con Python
- Familiarità con l'SDK Amazon SageMaker Python
- Familiarità con l'AWS Command Line Interface (AWS CLI)
- Familiarità con AWS SDK per Python (Boto3)
- Familiarità con Amazon ECR
- Familiarità con Docker

Versioni del prodotto

- SDK AWS Step Functions Data Science versione 2.3.0
- SDK Amazon SageMaker Python versione 2.78.0

Architecture

Il diagramma seguente mostra un esempio di flusso di lavoro per creare un'immagine di contenitore Docker per SageMaker, quindi utilizzarla per un modello di addestramento in Step Functions:

Il diagramma mostra il flusso di lavoro seguente:

1. Un data scientist o un DevOps ingegnere utilizza un SageMaker notebook Amazon per creare un'immagine di contenitore Docker personalizzata.
2. Un data scientist o un DevOps ingegnere archivia l'immagine del contenitore Docker in un repository privato Amazon ECR che si trova in un registro privato.
3. Un data scientist o un DevOps ingegnere utilizza il contenitore Docker per eseguire un processo di elaborazione SageMaker Python in un flusso di lavoro Step Functions.

Automazione e scalabilità

Il SageMaker notebook di esempio in questo modello utilizza un tipo di istanza di `ml.m5.xlarge` notebook. È possibile modificare il tipo di istanza in base al proprio caso d'uso. Per ulteriori informazioni sui tipi di istanze SageMaker notebook, consulta la pagina SageMaker dei prezzi di Amazon.

Tools (Strumenti)

- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- Amazon SageMaker è un servizio di machine learning (ML) gestito che ti aiuta a creare e addestrare modelli di machine learning per poi distribuirli in un ambiente ospitato pronto per la produzione.
- Amazon SageMaker Python SDK è una libreria open source per la formazione e la distribuzione di modelli di apprendimento automatico su. SageMaker
- AWS Step Functions è un servizio di orchestrazione serverless che ti aiuta a combinare le funzioni AWS Lambda e altri servizi AWS per creare applicazioni aziendali critiche.

- AWS Step Functions Data Science Python SDK è una libreria open source che ti aiuta a creare flussi di lavoro Step Functions che elaborano e pubblicano modelli di machine learning.

Epiche

Crea un'immagine di contenitore Docker personalizzata e archiviala in Amazon ECR

Operazione	Description	Competenze richieste
Configura Amazon ECR e crea un nuovo registro privato.	Se non l'hai già fatto, configura Amazon ECR seguendo le istruzioni in Configurazione con Amazon ECR nella Amazon ECR User Guide. Ogni account AWS è dotato di un registro Amazon ECR privato predefinito.	DevOps ingegnere
Crea un repository privato Amazon ECR.	Segui le istruzioni in Creazione di un repository privato nella Amazon ECR User Guide. **Note** Il repository che crei è il luogo in cui archivierai le immagini personalizzate dei contenitori Docker.	DevOps ingegnere
Crea un Dockerfile che includa le specifiche necessarie per eseguire il SageMaker processo di elaborazione.	Crea un Dockerfile che includa le specifiche necessarie per eseguire il SageMaker processo di elaborazione configurando un Dockerfile. Per istruzioni, consulta	DevOps ingegnere

Operazione	Description	Competenze richieste

	Adattamento del proprio contenitore di formazione nella Amazon SageMaker Developer Guide.
	
	Per ulteriori informazioni su Dockerfiles, consulta il Dockerfile Reference nella documentazione Docker.
	
	Esempio di celle di codice del notebook Jupyter per creare un Dockerfile
	
	Cella 1
	
	`# Make docker folder`
	`!mkdir -p docker`
	
	Cella 2
	
	`%%writefile docker/Do`
	`ckerfile`
	
	`FROM python:3.7-slim-bu`
	`ster`
	
	`RUN pip3 install`
	` pandas==0.25.3 scikit-`
	`learn==0.21.3`
	`ENV PYTHONUNBUFFERED=T`
	`RUE`
	
	`ENTRYPOINT ["python3"]`	

Operazione	Description	Competenze richieste
Crea l'immagine del tuo contenitore Docker e inviala ad Amazon ECR.	1. Crea l'immagine del contenitore utilizzando il Dockerfile che hai creato eseguendo il `docker build` comando nella CLI di AWS. 2. Invia l'immagine del contenitore ad Amazon ECR eseguendo il `docker push` comando. Per ulteriori informazioni, consulta Creazione e registrazione del contenitore in Costruire il proprio contenitore di algoritmi su. GitHub Esempio di celle di codice per notebook Jupyter per creare e registrare un'immagine Docker **Important** Prima di eseguire le seguenti celle, assicurati di aver creato un Dockerfile e di averlo archiviato nella directory chiamata. `docker` Inoltre, assicurati di aver creato un repository Amazon ECR e di sostituire il `ecr_repository`	DevOps ingegnere

Operazione	Description	Competenze richieste

	valore nella prima cella con il nome del repository.
	
	Cella 1
	
	`import boto3`
	`tag = ':latest'`
	`account_id = boto3.client('sts').get_caller_identity().get('Account')`
	`region = boto3.Session().region_name`
	`ecr_repository = 'byoc'`
	
	`image_uri = '{}.dkr.ecr.{}.amazonaws.com/{}'.format(account_id, region, ecr_repository + tag)`
	
	Cella 2
	
	`# Build docker image`
	`!docker build -t $image_uri docker`
	
	Cella 3
	
	`# Authenticate to ECR`
	`!aws ecr get-login-password --region {region} | docker login --username AWS --password-stdin`	

Operazione	Description	Competenze richieste

	`{account_id}.dkr.ecr.{region}.amazonaws.com`
	
	Cella 4
	
	`# Push docker image`
	`!docker push $image_uri`
	
	Note
	È necessario autenticare il client Docker nel registro privato in modo da poter utilizzare i comandi `docker push` and`docker pull`. Questi comandi inviano ed estraggono immagini da e verso gli archivi del registro.	

Crea un flusso di lavoro Step Functions che utilizzi l'immagine del contenitore Docker personalizzata

Operazione	Description	Competenze richieste
Crea uno script Python che includa la tua logica di elaborazione personalizzata e di addestramento dei modelli.	Scrivi una logica di elaborazione personalizzata da eseguire nello script di elaborazione dei dati. Quindi, salvalo come script Python denominato. `training.py`	Data Scientist

Operazione	Description	Competenze richieste

	Per ulteriori informazioni, consulta Bring your own model with SageMaker Script Mode on GitHub.
	
	Esempio di script Python che include l'elaborazione personalizzata e la logica di addestramento dei modelli
	
	`%%writefile training.py`
	`from numpy import empty`
	`import pandas as pd`
	`import os`
	`from sklearn import datasets, svm`
	`from joblib import dump, load`
	
	`if __name__ == '__main__':`
	`digits = datasets.load_digits()`
	`#create classifier object`
	`clf = svm.SVC(gamma=0.001, C=100.)`
	
	`#fit the model`
	`clf.fit(digits.data[:-1], digits.target[:-1])`
	
	`#model output in binary format`
	`output_path = os.path.join('/opt/`	

Operazione	Description	Competenze richieste
	ml/processing/model', "model.joblib") dump(clf, output_path)	

Operazione	Description	Competenze richieste

Crea un flusso di lavoro Step Functions che includa il tuo job SageMaker Processing come uno dei passaggi.	Installa e importa l'SDK AWS Step Functions Data Science e carica il file training.py su Amazon S3. Quindi, usa l'SDK Amazon SageMaker Python per definire una fase di elaborazione in Step Functions.
	
	Important
	Assicurati di aver creato un ruolo di esecuzione IAM per Step Functions nel tuo account AWS.
	
	Esempio di configurazione dell'ambiente e script di formazione personalizzato da caricare su Amazon S3
	
	`!pip install stepfunct ions`
	
	`import boto3`
	`import stepfunctions`
	`import sagemaker`
	`import datetime`
	
	`from stepfunctions import steps`
	`from stepfunct ions.inputs import ExecutionInput`
		Data Scientist

Operazione	Description	Competenze richieste

	from stepfunct
	ions.steps import (
	    Chain
	)
	from stepfunct
	ions.workflow import
	 Workflow
	from sagemaker
	.processing import
	 ScriptProcessor,
	 ProcessingInput,
	 ProcessingOutput
	
	sagemaker_session =
	 sagemaker.Session()
	bucket = sagemaker
	_session.default_b
	ucket()
	role = sagemaker
	.get_execution_role()
	prefix = 'byoc-tra
	ining-model'
	
	# See prerequisites
	 section to create this
	 role
	workflow_execution_rol
	e = f"arn:aws:iam::
	{account_id}:role/Ama
	zonSageMaker-StepF
	unctionsWorkflowEx
	ecutionRole"
	
	execution_input =
	 ExecutionInput(
	    schema={
	        "Preproce
	ssingJobName": str})
	
	input_code = sagemaker
	_session.upload_data(	

Operazione	Description	Competenze richieste

```
    "training.py",
    bucket=bucket,
    key_prefix="prepro
cessing.py",
)
```

Esempio di definizione della fase di SageMaker elaborazione che utilizza un'immagine Amazon ECR personalizzata e uno script Python

> **Note**
>
> Assicurati di utilizzare il `execution_input` parametro per specificare il nome del lavoro. Il valore del parametro deve essere univoco ogni volta che viene eseguito il processo. Inoltre, il codice del file training.py viene passato come `input` parametro a`ProcessingStep`, il che significa che verrà copiato all'interno del contenitore. La destinazione del `ProcessingInput` codice è la stessa del secondo argomento all'`container`

Operazione	Description	Competenze richieste

	`_entrypoint` interno di.
	
	`script_processor = ScriptProcessor(command=['python3'], image_uri=image_uri, role=role, instance_count=1, instance_type='ml.m5.xlarge')`
	
	`processing_step = steps.ProcessingStep( "training-step", processor=script_processor, job_name=execution_input["PreprocessingJobName"], inputs=[ ProcessingInput( source=input_code, destination="/opt/ml/processing/input/code", input_name="code", ), ], outputs=[`	

Operazione	Description	Competenze richieste

	<pre>Processin
	gOutput(
	source='/
	opt/ml/processing/
	model',
	destinati
	on="s3://{}/{}".fo
	rmat(bucket, prefix),
	output_na
	me='byoc-example')
	],
	container_entrypoi
	nt=["python3", "/opt/
	ml/processing/input/c
	ode/training.py"],
	)</pre> Esempio di flusso di lavoro Step Functions che esegue un SageMaker processo di elaborazione
	
	Note
	Questo flusso di lavoro di esempio include solo la fase del processo di SageMaker elaborazione, non un flusso di lavoro Step Functions completo. Per un esempio completo di flusso di lavoro, consulta Notebooks di esempio SageMaker nella documentazione dell'SDK AWS	

Operazione	Description	Competenze richieste

	Step Functions Data Science.
	
	`workflow_graph = Chain([processing_step])`
	
	`workflow = Workflow( name="ProcessingWorkflow", definition=workflow_graph, role=workflow_execution_role )`
	
	`workflow.create()`
	`# Execute workflow execution = workflow.execute( inputs={ "PreprocessingJobName": str(datetime.datetime.now().strftime("%Y%m%d%H%M-%SS")), # Each pre processing job (SageMaker processing job) requires a unique name, } )`
	`execution_output = execution.get_output(wait=True)`	

Risorse correlate

- Dati di processo (Amazon SageMaker Developer Guide)
- Adattamento del proprio contenitore di formazione (Amazon SageMaker Developer Guide)

Usa gli agenti Amazon Bedrock per automatizzare la creazione di controlli di accesso in Amazon EKS tramite istruzioni basate su testo

Keshav Ganesh e Sudhanshu Saurav, Amazon Web Services

Riepilogo

Le organizzazioni devono affrontare sfide nella gestione dei controlli degli accessi e nel provisioning delle risorse quando più team devono lavorare con un cluster Amazon Elastic Kubernetes Service (Amazon EKS) condiviso. Un servizio Kubernetes gestito come Amazon EKS ha semplificato le operazioni del cluster. Tuttavia, il sovraccarico amministrativo legato alla gestione dell'accesso del team e delle autorizzazioni alle risorse rimane complesso e richiede molto tempo.

Questo modello mostra come gli agenti Amazon Bedrock possono aiutarti ad automatizzare la gestione degli accessi ai cluster Amazon EKS. Questa automazione consente ai team di sviluppo di concentrarsi sullo sviluppo delle applicazioni principali anziché occuparsi della configurazione e della gestione del controllo degli accessi. Puoi personalizzare un agente Amazon Bedrock per eseguire azioni per un'ampia varietà di attività tramite semplici istruzioni in linguaggio naturale.

Utilizzando AWS Lambda le funzioni come gruppi di azione, un agente Amazon Bedrock può gestire attività come la creazione di voci di accesso degli utenti e la gestione delle politiche di accesso. Inoltre, un agente Amazon Bedrock può configurare associazioni di identità dei pod che consentono l'accesso alle risorse AWS Identity and Access Management (IAM) per i pod in esecuzione nel cluster. Utilizzando questa soluzione, le organizzazioni possono semplificare l'amministrazione dei cluster Amazon EKS con semplici istruzioni basate su testo, ridurre il sovraccarico manuale e migliorare l'efficienza complessiva dello sviluppo.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Ruoli e autorizzazioni IAM stabiliti per il processo di implementazione. Ciò include le autorizzazioni per accedere ai modelli Amazon Bedrock Foundation (FM), creare funzioni Lambda e qualsiasi altra risorsa richiesta nell'ambito della destinazione. Account AWS
- Accesso abilitato in modalità attiva Account AWS a questi Amazon Bedrock FMs: Amazon Titan Text Embeddings V2 e Anthropic Claude 3 Haiku.

- AWS Command Line Interface (AWS CLI) versione 2.9.11 o successiva, installata e configurata.
- eksctl 0.194.0 o versione successiva, installato.

Limitazioni

- Potrebbero essere necessarie formazione e documentazione per contribuire a garantire un'adozione agevole e un uso efficace di queste tecniche. L'uso di Amazon Bedrock, Amazon EKS, Lambda, OpenSearch Amazon Service e OpenAPI comporta una curva di apprendimento significativa per sviluppatori e team. DevOps
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità regionale, consulta i servizi AWS per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Architecture

Il diagramma seguente mostra i componenti del flusso di lavoro e dell'architettura per questo modello.

Questa soluzione esegue le seguenti operazioni:

1. L'utente interagisce con l'agente Amazon Bedrock inviando una richiesta o una query che funge da input per l'elaborazione e l'intervento dell'agente.
2. In base al prompt, l'agente Amazon Bedrock verifica lo schema OpenAPI per identificare l'API corretta da utilizzare come target. Se l'agente Amazon Bedrock trova la chiamata API corretta, la richiesta viene inviata al gruppo di azioni associato alla funzione Lambda che implementa queste azioni.
3. Se non viene trovata un'API pertinente, l'agente Amazon Bedrock interroga la OpenSearch raccolta. La OpenSearch raccolta utilizza contenuti indicizzati della knowledge base provenienti dal bucket Amazon S3 che contiene la Amazon EKS User Guide.
4. La OpenSearch raccolta restituisce informazioni contestuali pertinenti all'agente Amazon Bedrock.
5. Per le richieste utilizzabili (quelle che corrispondono a un'operazione API), l'agente Amazon Bedrock viene eseguito all'interno di un cloud privato virtuale (VPC) e attiva la funzione Lambda.
6. La funzione Lambda esegue un'azione basata sull'input dell'utente all'interno del cluster Amazon EKS.

7. Il bucket Amazon S3 per il codice Lambda memorizza l'artefatto contenente il codice e la logica scritti per la funzione Lambda.

Tools (Strumenti)

Servizi AWS

- Amazon Bedrock è un servizio completamente gestito che rende disponibili per l'uso modelli di base ad alte prestazioni (FMs) delle principali startup di intelligenza artificiale e di Amazon tramite un'API unificata.
- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su e. Account AWS Regioni AWS
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire AWS Kubernetes senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon OpenSearch Service è un servizio gestito che ti aiuta a distribuire, gestire e scalare OpenSearch i cluster in. Cloud AWS La sua funzionalità di raccolta ti aiuta a organizzare i tuoi dati e a creare basi di conoscenza complete che gli assistenti di intelligenza artificiale come gli agenti di Amazon Bedrock possono utilizzare.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri strumenti

- eksctl è un'utilità da riga di comando per la creazione e la gestione di cluster Kubernetes su Amazon EKS.

Repository di codice

Il codice per questo pattern è disponibile nel repository GitHub eks-access-controls-bedrock-agent.

Best practice

- Mantieni la massima sicurezza possibile durante l'implementazione di questo modello. Assicurati che il cluster Amazon EKS sia privato, abbia autorizzazioni di accesso limitate e che tutte le risorse siano all'interno di un cloud privato virtuale (VPC). Per ulteriori informazioni, consulta Best practice per la sicurezza nella documentazione di Amazon EKS.
- Usa le chiavi gestite AWS KMS dal cliente laddove possibile e concedi loro autorizzazioni di accesso limitate.
- Segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.

Epiche

Configura l'ambiente

Operazione	Description	Competenze richieste
Clonare il repository.	Per clonare il repository di questo pattern, esegui il seguente comando nella tua workstation locale: `git clone https://github.com/aws-samples/eks-access-controls-bedrock-agent.git`	AWS DevOps
Ottieni l' Account AWS ID.	Per ottenere l' Account AWS ID, procedi nel seguente modo: 1. Apri una shell nella cartella principale del repository clonato,. `eks-access-`	AWS DevOps

Operazione	Description	Competenze richieste

	`controls-bedrock-agent`
	2. Per ottenere il tuo Account AWS ID, vai alla directory clonata ed esegui il seguente comando:
	`AWS_ACCOUNT=$(aws sts get-caller-identity --query "Account" --output text)`
	Questo comando memorizza il tuo Account AWS ID nella `AWS_ACCOUNT` variabile.	

Operazione	Description	Competenze richieste

Crea il bucket S3 per il codice Lambda.	Per implementare questa soluzione, è necessario creare tre bucket Amazon S3 con scopi diversi, come mostrato nel diagramma di architettura. I bucket S3 sono per il codice Lambda, una knowledge base e lo schema OpenAPI.
	
	Per creare il bucket di codice Lambda, utilizza i seguenti passaggi:
	
	1. Per creare un bucket S3 per il codice Lambda, esegui il seguente comando:
	
	`aws s3 mb s3://bedrock-agent-lambda-artifacts-${AWS_ACCOUNT} --region us-east-1`
	
	2. Per installare la dipendenza dal codice Lambda, esegui il seguente comando:
	
	`cd eks-lambda`
	`npm install`
	`tsc`
	`cd .. && cd opensearch-lambda`
	`npm install`
	`tsc`
	`cd ..`
	
	3. Per impacchettare il codice e caricarlo nel bucket S3
		AWS DevOps

Operazione	Description	Competenze richieste
	per Lambda, esegui il seguente comando: `aws cloudformation package \ --template-file eks-access-controls.yaml \ --s3-bucket bedrock-agent-lambda-artifacts-${AWS_ACCOUNT} \ --output-template-file eks-access-controls-template.yaml \ --region us-east-1` Il comando package crea un nuovo CloudFormation modello (`eks-access-controls-template.yaml`) che contiene: • Riferimenti al codice della funzione Lambda memorizzato nel bucket S3. • Definizioni per tutta l' AWS infrastruttura richiesta, tra cui VPC, sottoreti, agente Amazon Bedrock e raccolta. OpenSearch Puoi utilizzare questo modello per distribuire la soluzione completa utilizzando. CloudFormation	

Operazione	Description	Competenze richieste

Crea il bucket S3 per la knowledge base.	Per creare il bucket Amazon S3 per la knowledge base, utilizza i seguenti passaggi:
	
	1. Per creare il bucket Amazon S3 per la knowledge base, esegui il seguente comando:
	
	`aws s3 mb s3://eks-knowledge-base-${AWS_ACCOUNT} --region us-east-1`
	
	2. Per scaricare la Amazon EKS User Guide e archiviar la in una directory, esegui i seguenti comandi:
	
	`mkdir dataSource`
	`cd dataSource`
	`curl https://docs.aws.amazon.com/pdfs/eks/latest/userguide/eks-ug.pdf -o eks-user-guide.pdf`
	
	3. Per caricare la guida per l'utente nel bucket S3 creato nel passaggio 1, esegui il seguente comando:
	
	`aws s3 cp eks-user-guide.pdf s3://eks-knowledge-base-${AWS_ACCOUNT} \`
		AWS DevOps

Operazione	Description	Competenze richieste

	`--region us-east-1 \`
	4. Per tornare alla directory principale, esegui il seguente comando:
	`cd ..`	

Crea il bucket S3 per lo schema OpenAPI.	Per creare il bucket Amazon S3 per lo schema OpenAPI, utilizza i seguenti passaggi:
	1. Per creare il bucket S3, esegui il seguente comando:
	`aws s3 mb s3://eks-openapi-schema-${AWS_ACCOUNT} --region us-east-1`
	2. Per caricare lo schema OpenAPI nel bucket S3, esegui il seguente comando:
	`aws s3 cp openapi-schema.yaml s3://eks-openapi-schema-${AWS_ACCOUNT} --region us-east-1`
		AWS DevOps

Implementa lo stack CloudFormation

Operazione	Description	Competenze richieste
Distribuisci lo CloudFormation stack.	Per distribuire lo CloudFormation stack, utilizzate il file CloudFormation modello creato in `eks-access-controls-template.yaml` precedenza. Per istruzioni più dettagliate, consulta Creare uno stack dalla CloudFormation console nella documentazione. CloudFormation **Note** Il provisioning dell' OpenSearch indice con il CloudFormation modello richiede circa 10 minuti. Dopo aver creato lo stack, prendi nota di `VPC_ID` e `PRIVATE_SUBNET ID` s.	AWS DevOps
Crea il cluster Amazon EKS.	Per creare il cluster Amazon EKS all'interno del VPC, utilizza i seguenti passaggi: 1. Crea una copia del file di `eks-config.yaml` configurazione e assegna alla copia il `eks-deploy.yaml` nome.	AWS DevOps

Operazione	Description	Competenze richieste

	2. Aprire `eks-deploy.yaml` in un editor di testo. Quindi, sostituisci i seguenti valori segnaposto con i valori dello stack distribuito:, , e `VPC_ID` `PRIVATE_SUBNET1` `PRIVATE_SUBNET2`
	3. Per creare il cluster utilizzando l'utilità eksctl, esegui il comando seguente:
	`eksctl create cluster -f eks-deploy.yaml`
	Note
	Il completamento di questo processo di creazione del cluster può richiedere fino a 15-20 minuti.
	4. Per verificare che il cluster sia stato creato correttamente, esegui i seguenti comandi:
	`aws eks describe-cluster --name --query "cluster.status"`
	`aws eks update-kubeconfig --name --region`	

Operazione	Description	Competenze richieste
	`kubectl get nodes` I risultati attesi sono i seguenti: • Lo stato del cluster è`ACTIVE`. • Il comando `kubectl get nodes` mostra che tutti i nodi sono in `Ready` stato.	

Connect la funzione Lambda e il cluster Amazon EKS

Operazione	Description	Competenze richieste
Crea una connessione tra il cluster Amazon EKS e la funzione Lambda.	Per configurare le autorizzazioni di rete e IAM per consentire alla funzione Lambda di comunicare con il cluster Amazon EKS, utilizza i seguenti passaggi: 1. Per identificare il ruolo IAM associato alla funzione Lambda, apri Console di gestione AWS e individua la funzione Lambda denominata. `bedrock-agent-eks-access-control` Prendi nota dell'Amazon Resource Name (ARN) del ruolo IAM. 2. Per creare una voce di accesso nel cluster Amazon EKS per il ruolo IAM della	AWS DevOps

Operazione	Description	Competenze richieste

	funzione Lambda, esegui il comando seguente:
	
	`aws eks create-access-entry --cluster-name eks-testing-cluster --principal-arn <principal-Role-ARN>`
	
	3. Per assegnare `AmazonEKSClusterAdminPolicy` le autorizzazioni a questo ruolo, esegui il seguente comando:
	
	`aws eks associate-access-policy --cluster-name eks-testing-cluster --principal-arn <principal-Role-ARN> --policy-arn arn:aws:eks::aws:cluster-access-policy/AmazonEKSClusterAdminPolicy --access-scope type=cluster`
	
	Per ulteriori informazioni, consulta Associare le politiche di accesso alle voci di accesso e Amazon EKSCluster AdminPolicy nella documentazione di Amazon EKS.	

Operazione	Description	Competenze richieste
	4. Individua il gruppo di sicurezza del cluster Amazon EKS. Aggiungi una regola in entrata per consentire il traffico di rete in entrata dalla funzione Lambda al cluster Amazon EKS. Utilizza i seguenti valori per la regola in entrata: • Tipo: HTTPS • Intervallo di porte: 443 • Fonte: gruppo di sicurezza Lambda Per ulteriori informazioni, consulta Configurare le regole dei gruppi di sicurezza nella documentazione di Amazon VPC.	

Test della soluzione

Operazione	Description	Competenze richieste
Prova l'agente Amazon Bedrock.	Prima di testare l'agente Amazon Bedrock, assicurati di effettuare le seguenti operazioni: • Esegui prima il test con ruoli non di produzione.	AWS DevOps

Operazione	Description	Competenze richieste
	• Documenta tutte le modifiche apportate all'accesso al cluster. • Predisponi di un piano per annullare le modifiche, se necessario. Per accedere all'agente Amazon Bedrock, utilizza i seguenti passaggi: 1. Accedi all' Console di gestione AWS utilizzo di un ruolo IAM con autorizzazioni Amazon Bedrock e apri la console Amazon Bedrock all'indirizzo. https://console.aws.amazon.com/bedrock/ 2. Seleziona Agenti nel riquadro di navigazione a sinistra. Quindi, scegli l'agente configurato nella sezione Agenti. 3. Per testare l'agente, prova i seguenti prompt di esempio in cui sostituisci l'ARN `Principal-ARN-OF-ROLE` con un ruolo IAM effettivo: • Per creare una voce di accesso per qualsiasi ruolo IAM a cui desideri fornire	

Operazione	Description	Competenze richieste
	l'accesso al cluster EKS, utilizza il seguente prompt: `Create an access entry in cluster eks-testing-new for a role whose principal arn is <Principal-ARN-OF-ROLE> with access policy as AmazonEKSAdminPolicy` Risultato previsto: • L'agente deve confermare la creazione della voce di accesso. • Per verificare, verifica l'utilizzo Console di gestione AWS o utilizza l'API Amazon EKS ed esegui il seguente comando: `aws eks list-access-entries --cluster-name ekscluster` • Per descrivere la voce di accesso che hai creato, usa il seguente prompt: `Describe an access entry in cluster eks-testing-new whose principal arn is <Principal-ARN-OF-ROLE>`	

Operazione	Description	Competenze richieste
	Risultato previsto: • L'agente deve restituire i dettagli sulla voce di accesso. • I dettagli devono corrispondere a quelli configurati in precedenza per la voce di accesso. • Per eliminare la voce di accesso creata, utilizzate il seguente prompt: `Delete the access entry in cluster eks-testing-new whose principal arn is <Principal-ARN-OF-ROLE>` Risultato previsto: • L'agente deve confermare l'eliminazione della voce di accesso. • Per verificare, verifica l'utilizzo Console di gestione AWS o utilizza l'API Amazon EKS ed esegui il seguente comando: `aws eks list-access-entries --cluster-name ekscluster` Puoi anche chiedere all'agente di eseguire azioni per le	

Operazione	Description	Competenze richieste
	associazioni EKS Pod Identity. Per ulteriori dettagli, consulta Scopri come EKS Pod Identity concede l'accesso ai pod Servizi AWS nella documentazione di Amazon EKS.	

Eliminazione

Operazione	Description	Competenze richieste
Eliminare le risorse.	Per ripulire le risorse create da questo modello, utilizzare la procedura seguente. Attendi il completamento di ogni passaggio di eliminazione prima di procedere al passaggio successivo. **Warning** Questa procedura eliminerà definitivamente tutte le risorse create da questi stack. Assicurati di aver eseguito il backup di tutti i dati importanti prima di procedere. 1. Per eliminare il cluster Amazon EKS, esegui il seguente comando:	AWS DevOps

Operazione	Description	Competenze richieste

	`eksctl delete cluster -f eks-deploy.yaml`
	
	Note
	Il completamento di questa operazione può richiedere 15-20 minuti.
	
	2. Per eliminare i bucket Amazon S3, esegui i seguenti comandi:
	• Per svuotare il secchio Lambda:
	`aws s3 rm s3://bedrock-agent-lambda-artifacts-${AWS_ACCOUNT} --recursive`
	• Per svuotare il bucket della knowledge base:
	`aws s3 rm s3://eks-knowledge-base-${AWS_ACCOUNT} –recursive`
	• Per svuotare il bucket dello schema OpenAPI:
	`aws s3 rm s3://bedrock-agent-openapi-schema-${AWS_ACCOUNT} –recursive`	

Operazione	Description	Competenze richieste

	• Per eliminare i bucket vuoti:
	`aws s3 rb s3://bedrock-agent-lambda-artifacts-${AWS_ACCOUNT}`
	`aws s3 rb s3://eks-knowledge-base-${AWS_ACCOUNT}`
	`aws s3 rb s3://bedrock-agent-openapi-schema-${AWS_ACCOUNT}`
	3. Per eliminare lo CloudFormation stack, esegui il seguente comando:
	`aws cloudformation delete-stack \ --stack-name`
	4. Per verificare l'eliminazione del cluster Amazon EKS, esegui il seguente comando:
	`eksctl get clusters`
	5. Per verificare l'eliminazione dei bucket Amazon S3, esegui i seguenti comandi:
	• Per verificare l'eliminazione del bucket Lambda:	

Operazione	Description	Competenze richieste

	`aws s3 ls | grep "bedrock-agent-lambda-artifacts"`
	• Per verificare l'eliminazione del bucket della knowledge base:
	`aws s3 ls | grep "eks-knowledge-base"`
	• Per verificare l'eliminazione del bucket dello schema OpenAPI:
	`aws s3 ls | grep "bedrock-agent-openapi-schema"`
	6. Per verificare l'eliminazione dello stack, esegui il seguente comando:
	`aws cloudformation list-stacks \--query 'StackSummaries[?StackName==``]'`
	Se lo stack non viene eliminato, vedi Risoluzione dei problemi.	

risoluzione dei problemi

Problema	Soluzione
Durante la configurazione dell'ambiente viene restituito un codice di errore diverso da zero.	Verifica di utilizzare la cartella corretta quando esegui un comando per distribuire questa soluzione. Per ulteriori informazioni, consultat e il file First_Deploy.md nel repository di questo pattern.
La funzione Lambda non è in grado di eseguire l'operazione.	Assicurati che la connettività sia configurata correttamente dalla funzione Lambda al cluster Amazon EKS.
I prompt dell'agente non riconoscono il. APIs	Ridistribuisci la soluzione. Per ulteriori informazi oni, consultate il file RE_Deploy.md nel repository di questo pattern.
Lo stack non viene eliminato.	Un tentativo iniziale di eliminare lo stack potrebbe fallire. Questo errore può verificar si a causa di problemi di dipendenza con la risorsa personalizzata creata per la OpenSearc h raccolta che esegue l'indicizzazione per la knowledge base. Per eliminare lo stack, ritenta l'operazione di eliminazione conservando la risorsa personalizzata.

Risorse correlate

AWS Blog

- Un'analisi approfondita dei controlli semplificati di gestione degli accessi di Amazon EKS

Documentazione Amazon Bedrock

- Automatizza le attività nella tua applicazione utilizzando agenti AI
- Come funziona Amazon Bedrock Agents

- Verifica e risolvi i problemi del comportamento degli agenti
- Utilizza i gruppi di azioni per definire le azioni che il tuo agente deve eseguire

Documentazione Amazon EKS

- Learn how access control works in Amazon EKS

Implementa uno use case RAG AWS utilizzando Terraform e Amazon Bedrock

Martin Maritsch, Nicolas Jacob Baer, Olivier Brique, Julian Ferdinand Grueber, Alice Morano e Nicola D Orazio, Amazon Web Services

Riepilogo

AWS offre varie opzioni per creare casi d'uso di intelligenza artificiale generativa abilitati per Retrieval Augmented Generation (RAG). Questo modello fornisce una soluzione per un'applicazione basata su RAG basata su Amazon Aurora LangChain e compatibile con PostgreSQL come archivio vettoriale. Puoi implementare direttamente questa soluzione con Terraform nel tuo e implementare il seguente semplice caso d'uso RAG: Account AWS

1. L'utente carica manualmente un file in un bucket Amazon Simple Storage Service (Amazon S3), come un file Microsoft Excel o un documento PDF. (Per ulteriori informazioni sui tipi di file supportati, consulta la documentazione non strutturata.)
2. Il contenuto del file viene estratto e incorporato in un database di conoscenze basato su Aurora PostgreSQL senza server, compatibile con Aurora, che supporta l'inserimento quasi in tempo reale di documenti nell'archivio vettoriale. Questo approccio consente al modello RAG di accedere e recuperare le informazioni pertinenti per i casi d'uso in cui le basse latenze sono importanti.
3. Quando l'utente interagisce con il modello di generazione del testo, migliora l'interazione attraverso il recupero e l'aumento dei contenuti pertinenti dai file precedentemente caricati.

Il pattern utilizza Amazon Titan Text Embeddings v2 come modello di incorporamento e Anthropic Claude 3 Sonnet come modello di generazione del testo, entrambi disponibili su Amazon Bedrock.

Prerequisiti e limitazioni

Prerequisiti

- Account AWS Un attivo.
- AWS Command Line Interface (AWS CLI) installato e configurato con il tuo Account AWS. Per istruzioni sull'installazione, consulta Installazione o aggiornamento alla versione più recente di AWS CLI nella AWS CLI documentazione. Per rivedere AWS le tue credenziali e l'accesso al tuo account, consulta Configurazione e impostazioni dei file di credenziali nella AWS CLI documentazione.

- Accesso ai modelli abilitato per i modelli linguistici di grandi dimensioni richiesti (LLMs) nella tua Account AWS console Amazon Bedrock. Questo modello richiede quanto segue: LLMs
  - `amazon.titan-embed-text-v2:0`
  - `anthropic.claude-3-sonnet-20240229-v1:0`

Limitazioni

- Questa architettura di esempio non include un'interfaccia per la risposta programmatica alle domande con il database vettoriale. Se il tuo caso d'uso richiede un'API, valuta la possibilità di aggiungere Amazon API Gateway con una AWS Lambda funzione che esegue attività di recupero e risposta alle domande.
- Questa architettura di esempio non include funzionalità di monitoraggio per l'infrastruttura distribuita. Se il tuo caso d'uso richiede il monitoraggio, prendi in considerazione l'aggiunta di servizi AWS di monitoraggio.
- Se carichi molti documenti in un breve lasso di tempo nel bucket Amazon S3, la funzione Lambda potrebbe incontrare dei limiti di velocità. Come soluzione, puoi disaccoppiare la funzione Lambda con una coda Amazon Simple Queue Service (Amazon SQS) in cui puoi controllare la frequenza delle chiamate Lambda.
- Alcuni Regioni AWS non sono disponibili in tutti. Servizi AWS Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- AWS CLI versione 2 o successiva
- Docker versione 26.0.0 o successiva
- Poetry versione 1.7.1 o successiva
- Python versione 3.10 o successiva
- Terraform versione 1.8.4 o successiva

Architecture

Il diagramma seguente mostra i componenti del flusso di lavoro e dell'architettura per questo modello.

Questo diagramma illustra quanto segue:

1. Quando un oggetto viene creato nel bucket Amazon S3`bedrock-rag-template-<account_id>`, una notifica Amazon S3 richiama la funzione Lambda. `data-ingestion-processor`
2. La funzione Lambda si `data-ingestion-processor` basa su un'immagine Docker memorizzata nel repository Amazon Elastic Container Registry (Amazon ECR). `bedrock-rag-template`

   La funzione utilizza LangChain S3 FileLoader per leggere il file come documento. LangChain Quindi, i LangChain RecursiveCharacterTextSplitterblocchi di ogni documento, con una `CHUNK_SIZE` e una `CHUNK_OVERLAP` che dipendono dalla dimensione massima del token del modello di incorporamento Amazon Titan Text Embedding V2. Successivamente, la funzione Lambda richiama il modello di incorporamento su Amazon Bedrock per incorporare i blocchi in rappresentazioni vettoriali numeriche. Infine, questi vettori sono archiviati nel database Aurora PostgreSQL. Per accedere al database, la funzione Lambda recupera innanzitutto il nome utente e la password da. Gestione dei segreti AWS
3. Sull'istanza del notebook Amazon SageMaker AIaws-sample-bedrock-rag-template, l'utente può scrivere un prompt di domande. Il codice richiama Claude 3 su Amazon Bedrock e aggiunge le informazioni della knowledge base al contesto del prompt. Di conseguenza, Claude 3 fornisce risposte utilizzando le informazioni contenute nei documenti.

L'approccio di questo modello alla rete e alla sicurezza è il seguente:

- La funzione Lambda `data-ingestion-processor` si trova in una sottorete privata all'interno del cloud privato virtuale (VPC). La funzione Lambda non è autorizzata a inviare traffico alla rete Internet pubblica a causa del suo gruppo di sicurezza. Di conseguenza, il traffico verso Amazon S3 e Amazon Bedrock viene instradato solo attraverso gli endpoint VPC. Di conseguenza, il traffico non attraversa la rete Internet pubblica, il che riduce la latenza e aggiunge un ulteriore livello di sicurezza a livello di rete.
- Tutte le risorse e i dati vengono crittografati laddove applicabile utilizzando la chiave AWS Key Management Service (AWS KMS) con l'alias. `aws-sample/bedrock-rag-template`

Automazione e scalabilità

Questo modello utilizza Terraform per distribuire l'infrastruttura dal repository di codice in un. Account AWS

Tools (Strumenti)

Servizi AWS

- Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL. In questo modello, Aurora PostgreSQL Compatible utilizza il plugin pgvector come database vettoriale.
- Amazon Bedrock è un servizio completamente gestito che rende disponibili per l'uso modelli di base ad alte prestazioni (FMs) delle principali startup di intelligenza artificiale e di Amazon tramite un'API unificata.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire Servizi AWS tramite comandi nella shell della riga di comando.
- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile. In questo modello, Amazon ECR ospita l'immagine Docker per la funzione Lambda`data-ingestion-processor`.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi. In questo modello, Lambda inserisce i dati nell'archivio vettoriale.
- Amazon SageMaker AI è un servizio di machine learning (ML) gestito che ti aiuta a creare e addestrare modelli di machine learning per poi distribuirli in un ambiente ospitato pronto per la produzione.
- Con Gestione dei segreti AWS puoi sostituire le credenziali nel codice,incluse le password, con una chiamata API a Secrets Manager in modo da recuperare il segreto a livello di codice.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS. Il VPC include sottoreti e tabelle di routing per controllare il flusso di traffico.

Altri strumenti

- Docker è un insieme di prodotti Platform as a Service (PaaS) che utilizzano la virtualizzazione a livello di sistema operativo per fornire software in container.
- HashiCorp Terraform è uno strumento Infrastructure as Code (IaC) che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud.
- Poetry è uno strumento per la gestione delle dipendenze e la creazione di pacchetti in Python.
- Python è un linguaggio di programmazione per computer generico.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub terraform-rag-template-using-amazon-bedrock.

Best practice

- Sebbene questo esempio di codice possa essere distribuito in qualsiasi ambiente Regione AWS, ti consigliamo di utilizzare Stati Uniti orientali (Virginia settentrionale) `us-east-1` o Stati Uniti occidentali (California settentrionale). `us-west-1` Questa raccomandazione si basa sulla disponibilità di modelli di base e di incorporamento in Amazon Bedrock al momento della pubblicazione di questo modello. Per un up-to-date elenco dei modelli Amazon Bedrock Foundation supportati in Regioni AWS, consulta Model support by Regione AWS nella documentazione di Amazon Bedrock. Per informazioni sulla distribuzione di questo esempio di codice in altre regioni, consulta Informazioni aggiuntive.
- Questo modello fornisce solo una demo proof-of-concept (PoC) o pilota. Se vuoi portare il codice in produzione, assicurati di utilizzare le seguenti best practice:
  - Abilita la registrazione degli accessi al server per Amazon S3.
  - Imposta il monitoraggio e gli avvisi per la funzione Lambda.
  - Se il tuo caso d'uso richiede un'API, prendi in considerazione l'aggiunta di Amazon API Gateway con una funzione Lambda che esegue attività di recupero e risposta alle domande.
- Segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.

Epiche

Implementa la soluzione in un Account AWS

Operazione	Description	Competenze richieste
Clonare il repository.	Per clonare il GitHub repository fornito con questo pattern, utilizzate il seguente comando: `git clone https://github.com/aws-samples/terraform-rag-template-using-amazon-bedrock`	AWS DevOps
Configura le variabili.	Per configurare i parametri per questo pattern, procedi come segue: 1. Sul tuo computer, nel GitHub repository, usa il seguente comando per aprire la `terraform` cartella: `cd terraform` 2. Apri il `commons.tfvars` file e personalizza i parametri in base alle tue esigenze.	AWS DevOps
Distribuire la soluzione.	Per distribuire la soluzione, procedi come segue: 1. Nella `terraform` cartella, usa il seguente comando per eseguire Terraform e	AWS DevOps

Operazione	Description	Competenze richieste

	passare le variabili che hai personalizzato:
	
	`terraform init`
	`terraform apply -var-file=commons.tfvars`
	
	2. Verifica che le risorse mostrate nel diagramma di architettura siano state distribuite correttamente.
	
	L'implementazione dell'infrastruttura fornisce un'istanza SageMaker AI all'interno del VPC e con le autorizzazioni per accedere al database Aurora PostgreSQL.	

Test della soluzione

Operazione	Description	Competenze richieste
Esegui la demo.	Una volta completata la precedente installazione dell'infrastruttura, segui i seguenti passaggi per eseguire la demo su un notebook Jupyter: 1. Accedi alla pagina Console di gestione AWS del luogo in Account AWS cui è distribuita l'infrastruttura.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	2. Apri l'istanza `aws-sample-bedrock-rag-template` del notebook SageMaker AI. 3. Sposta il notebook `rag_demo.ipynb` Jupyter sull'istanza del notebook SageMaker AI trascinandolo. 4. Apri l'`rag_demo.ipynb` istanza del notebook SageMaker AI e scegli il kernel. `conda_python3` 5. Per eseguire la demo, esegui le celle del notebook. Il notebook Jupyter ti guida attraverso il seguente processo: • Requisiti di installazione • Definizione di incorporamento • Connessioni database • Inserimento di dati • Recupero e generazione di testo aumentata • Domande relative ai documenti pertinenti	

Pulisci l'infrastruttura

Operazione	Description	Competenze richieste
Pulisci l'infrastruttura.	Per rimuovere tutte le risorse che hai creato quando non sono più necessarie, usa il seguente comando: `terraform destroy -var-file=commons.tfvars`	AWS DevOps

Risorse correlate

AWS resources

- Creazione di funzioni Lambda con Python
- Parametri di inferenza per i modelli di base
- Accesso ai modelli Amazon Bedrock Foundation
- Il ruolo dei database vettoriali nelle applicazioni di intelligenza artificiale generativa (AWS Database Blog)
- Lavorare con Amazon Aurora PostgreSQL

Altre risorse

- documentazione pgvector

Informazioni aggiuntive

Implementazione di un database vettoriale

Questo modello utilizza la compatibilità con Aurora PostgreSQL per implementare un database vettoriale per RAG. In alternativa ad Aurora PostgreSQL, AWS offre altre funzionalità e servizi per RAG, come Amazon Bedrock Knowledge Bases e Amazon Service. OpenSearch Puoi scegliere la soluzione più adatta alle tue esigenze specifiche:

- Amazon OpenSearch Service fornisce motori di ricerca e analisi distribuiti che puoi utilizzare per archiviare e interrogare grandi volumi di dati.
- Amazon Bedrock Knowledge Bases è progettato per creare e distribuire basi di conoscenza come astrazione aggiuntiva per semplificare il processo di acquisizione e recupero di RAG. Le Knowledge Base di Amazon Bedrock possono funzionare sia con Aurora PostgreSQL che con Amazon Service. OpenSearch

Distribuzione su altri Regioni AWS

Come descritto in Architettura, si consiglia di utilizzare la regione Stati Uniti orientali (Virginia settentrionale) `us-east-1` o Stati Uniti occidentali (California settentrionale) `us-west-1` per distribuire questo esempio di codice. Tuttavia, esistono due modi possibili per distribuire questo esempio di codice in regioni diverse da and. `us-east-1` `us-west-1` È possibile configurare la regione di distribuzione nel `commons.tfvars` file. Per l'accesso al modello di base interregionale, considera le seguenti opzioni:

- Attraversamento della rete Internet pubblica: se il traffico può attraversare la rete Internet pubblica, aggiungi gateway Internet al VPC. Quindi, modifica il gruppo di sicurezza assegnato alla funzione Lambda `data-ingestion-processor` e all'istanza del notebook SageMaker AI per consentire il traffico in uscita verso la rete Internet pubblica.
- Non utilizzare la rete Internet pubblica: per distribuire questo esempio in una regione diversa da `us-east-1` o`us-west-1`, procedi come segue:

1. Nella `us-west-1` regione `us-east-1` o, crea un VPC aggiuntivo che includa un endpoint VPC per. `bedrock-runtime`
2. Crea una connessione peering utilizzando il peering VPC o un gateway di transito verso il VPC dell'applicazione.
3. Quando configuri il client `bedrock-runtime` boto3 in qualsiasi funzione Lambda al di fuori di `us-east-1` o`us-west-1`, passa il nome DNS privato dell'endpoint VPC per `bedrock-runtime` in `us-east-1` o us-west-1 come nome al client boto3. `endpoint_url`

Implementa la logica di preelaborazione in un modello ML in un singolo endpoint utilizzando una pipeline di inferenza in Amazon SageMaker

Mohan Gowda Purushothama, Gabriel Rodriguez Garcia e Mateusz Zaremba, Amazon Web Services

Riepilogo

Questo modello spiega come distribuire più oggetti del modello di pipeline in un singolo endpoint utilizzando una pipeline di inferenza in Amazon. SageMaker L'oggetto del modello di pipeline rappresenta diverse fasi del flusso di lavoro di machine learning (ML), come la preelaborazione, l'inferenza del modello e la postelaborazione. Per illustrare la distribuzione di oggetti del modello di pipeline connessi in serie, questo modello mostra come implementare un contenitore Scikit-learn di preelaborazione e un modello di regressione basato sull'algoritmo di apprendimento lineare integrato. SageMaker L'implementazione SageMaker è ospitata dietro un singolo endpoint in.

> **Note**
>
> La distribuzione in questo modello utilizza il tipo di istanza ml.m4.2xlarge. Ti consigliamo di utilizzare un tipo di istanza in linea con i requisiti di dimensione dei dati e la complessità del flusso di lavoro. Per ulteriori informazioni, consulta la pagina SageMaker dei prezzi di Amazon. Questo modello utilizza immagini Docker predefinite per Scikit-learn, ma puoi usare i tuoi contenitori Docker e integrarli nel tuo flusso di lavoro.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Python 3.9
- SDK Amazon SageMaker Python e libreria Boto3
- Ruolo AWS Identity and Access Management (AWS IAM) con SageMaker autorizzazioni di base e autorizzazioni Amazon Simple Storage Service (Amazon S3)

Versioni del prodotto

- SDK Amazon SageMaker Python 2.49.2

Architecture

Stack tecnologico Target

- Amazon Elastic Container Registry (Amazon ECR)
- Amazon SageMaker
- Amazon SageMaker Studio
- Amazon Simple Storage Service (Amazon S3)
- Endpoint di inferenza in tempo reale per Amazon SageMaker

Architettura Target

Il diagramma seguente mostra l'architettura per la distribuzione di un oggetto del modello Amazon SageMaker Pipeline.

Il diagramma mostra il flusso di lavoro seguente:

1. Un SageMaker notebook implementa un modello di pipeline.
2. Un bucket S3 memorizza gli artefatti del modello.
3. Amazon ECR ottiene le immagini del contenitore di origine dal bucket S3.

Tools (Strumenti)

Strumenti AWS

- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- Amazon SageMaker è un servizio di machine learning gestito che ti aiuta a creare e addestrare modelli di machine learning per poi distribuirli in un ambiente ospitato pronto per la produzione.
- Amazon SageMaker Studio è un ambiente di sviluppo integrato (IDE) basato sul Web per il machine learning che ti consente di creare, addestrare, eseguire il debug, distribuire e monitorare i tuoi modelli di machine learning.

- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Codice

Il codice per questo pattern è disponibile nel repository GitHub Inference Pipeline con Scikit-learn e Linear Learner.

Epiche

Prepara il set di dati

Operazione	Description	Competenze richieste

Prepara il set di dati per l'attività di regressione.	Apri un notebook in Amazon SageMaker Studio.
	
	Per importare tutte le librerie necessarie e inizializzare l'ambiente di lavoro, usa il seguente codice di esempio nel tuo notebook:
	
	`import sagemaker`
	`from sagemaker import get_execution_role`
	
	`sagemaker_session = sagemaker.Session()`
	
	`# Get a SageMaker-compatible role used by this Notebook Instance.`
	`role = get_execution_role()`
	
	`# S3 prefix`
	`bucket = sagemaker_session.default_bucket()`
		Data Scientist

Operazione	Description	Competenze richieste

	`prefix = "Scikit-L inearLearner-pipel ine-abalone-example"`
	
	Per scaricare un set di dati di esempio, aggiungi il seguente codice al tuo taccuino:
	
	`! mkdir abalone_data`
	`! aws s3 cp s3://sage maker-sample-files /datasets/tabular/ uci_abalone/abalon e.csv ./abalone_data`
	
	Note
	L'esempio in questo modello utilizza l'Abalone Data Set dell'UCI Machine Learning Repository.	

Operazione	Description	Competenze richieste

Carica il set di dati in un bucket S3.	Nel taccuino in cui hai preparato il set di dati in precedenza, aggiungi il codice seguente per caricare i dati di esempio in un bucket S3:
	`WORK_DIRECTORY = "abalone_data"`
	`train_input = sagemaker_session.upload_data( path="{}/{}".format(WORK_DIRECTORY, "abalone.csv"), bucket=bucket, key_prefix="{}/{}".format(prefix, "train"), )`
		Data Scientist

Crea il preprocessore di dati utilizzando SKLearn

Operazione	Description	Competenze richieste
Preparare lo script preprocessor.py.	1. Copia la logica di preelaborazione dal file Python nel GitHub repository sklearn_abalone_featurizer.py, quindi incolla il codice in un file Python separato chiamato. `sklearn_abalone_featurizer.py` È possibile modificare il codice per adattarlo al set di dati e al flusso di lavoro personalizzati.	Data Scientist

Operazione	Description	Competenze richieste
	2. Salvate il `sklearn_abalone_featurizer.py` file nella directory principale del progetto (ovvero nella stessa posizione in cui eseguite il SageMaker notebook).	

Operazione	Description	Competenze richieste

Create l'oggetto SKLearn del preprocessore.	Per creare un oggetto SKLearn preprocessore (chiamato SKLearn Estimator) da incorporare nella pipeline di inferenza finale, esegui il codice seguente nel tuo taccuino: SageMaker
	`from sagemaker.sklearn. estimator import SKLearn`
	
	`FRAMEWORK_VERSION = "0.23-1" script_path = "sklearn_abalone_f eaturizer.py"`
	
	`sklearn_preprocessor = SKLearn( entry_point=script _path, role=role, framework_version= FRAMEWORK_VERSION, instance_type="ml. c4.xlarge", sagemaker_session= sagemaker_session, ) sklearn_preproc essor.fit({"train": train_input})`
		Data Scientist

Operazione	Description	Competenze richieste

Verifica l'inferenza del preprocessore.	Per confermare che il preprocessore sia definito correttamente, avviate un processo di trasformazione in batch inserendo il seguente codice nel notebook: SageMaker
	
	`# Define a SKLearn Transformer from the trained SKLearn Estimator`
	`transformer = sklearn_preprocessor.transformer(`
	`instance_count=1, instance_type="ml.m5.xlarge", assemble_with="Line", accept="text/csv"`
	`)`
	
	`# Preprocess training input`
	`transformer.transform(train_input, content_type="text/csv")`
	`print("Waiting for transform job: " + transformer.latest_transform_job.job_name)`
	`transformer.wait()`
	`preprocessed_train = transformer.output_path`	

Crea un modello di apprendimento automatico

Operazione	Description	Competenze richieste

Crea un oggetto modello.	Per creare un oggetto modello basato sull'algoritmo Linear Learner, inserisci il seguente codice nel tuo SageMaker taccuino:
	`import boto3`
	`from sagemaker`
	`.image_uris import`
	` retrieve`
	
	`ll_image = retrieve(`
	`"linear-learner",`
	` boto3.Session().re`
	`gion_name)`
	`s3_ll_output_key`
	`_prefix = "ll_train`
	`ing_output"`
	`s3_ll_output_location`
	` = "s3://{}/{}/{}/{}"`
	`.format(`
	`  bucket, prefix,`
	` s3_ll_output_key_p`
	`refix, "ll_model"`
	`)`
	
	`ll_estimator =`
	` sagemaker.estimato`
	`r.Estimator(`
	`  ll_image,`
	`  role,`
	`  instance_count=1,`
	`  instance_type="ml.`
	`m4.2xlarge",`
	`  volume_size=20,`
	`  max_run=3600,`
	`  input_mode="File",`
		Data Scientist

Operazione	Description	Competenze richieste

	`output_path=s3_ll_output_location, sagemaker_session=sagemaker_session, )`
	
	`ll_estimator.set_hyperparameters(feature_dim=10, predictor_type="regressor", mini_batch_size=32)`
	
	`ll_train_data = sagemaker.inputs.TrainingInput( preprocessed_train, distribution="FullyReplicated", content_type="text/csv", s3_data_type="S3Prefix", )`
	
	`data_channels = {"train": ll_train_data} ll_estimator.fit(inputs=data_channels, logs=True)`
	
	Il codice precedente recupera l'immagine Docker Amazon ECR pertinente dal registro Amazon ECR pubblico per il modello, crea un oggetto estimatore e quindi utilizza	

Operazione	Description	Competenze richieste
	quell'oggetto per addestrare il modello di regressione.	

Implementa la pipeline finale

Operazione	Description	Competenze richieste

Implementa il modello di pipeline.	Per creare un oggetto del modello di pipeline (ovvero un oggetto preprocessore) e distribuire l'oggetto, inserite il codice seguente nel vostro taccuino: SageMaker
	`from sagemaker.model import Model`
	`from sagemaker.pipeline import PipelineModel`
	`import boto3`
	`from time import gmtime, strftime`
	`timestamp_prefix = strftime("%Y-%m-%d-%H-%M-%S", gmtime())`
	`scikit_learn_inferencee_model = sklearn_preprocessor.create_model()`
	`linear_learner_model = ll_estimator.create_model()`
	`model_name = "inference-pipeline-" + timestamp_prefix`
		Data Scientist

Operazione	Description	Competenze richieste

	`endpoint_name = "inference-pipeline-ep-" + timestamp_prefix sm_model = PipelineModel( name=model_name, role=role, models= [scikit_learn_inferencee_model, linear_learner_model] )`
	
	`sm_model.deploy(initial_instance_count =1, instance_type="ml.c4.xlarge", endpoint_name=endpoint_name)`
	
	Note
	È possibile modificare il tipo di istanza utilizzato nell'oggetto del modello in base alle proprie esigenze.	

Operazione	Description	Competenze richieste

Verificate l'inferenza.	Per confermare che l'endpoint funzioni correttamente, esegui il seguente codice di inferenza di esempio sul tuo notebook: SageMaker
	
	`from sagemaker.predicto r import Predictor from sagemaker.serializ ers import CSVSerial izer`
	
	`payload = "M, 0.44, 0.365, 0.125, 0.516, 0.2155, 0.114, 0.155" actual_rings = 10 predictor = Predictor( endpoint_name=endp oint_name, sagemaker _session=sagemaker _session, serialize r=CSVSerializer() )`
	
	`print(predictor .predict(payload))`
		Data Scientist

Risorse correlate

- Preelabora i dati di input prima di fare previsioni utilizzando le pipeline di SageMaker inferenza di Amazon e Scikit-learn (AWS Machine Learning Blog)
- Machine Learning end-to-end con Amazon SageMaker (GitHub)

Implementa la convalida della sicurezza della codifica in tempo reale utilizzando un server MCP con Kiro e altri assistenti di codifica

Ivan Girardi e Iker Reina Fuente, Amazon Web Services

Riepilogo

Questo modello descrive come implementare un server Model Context Protocol (MCP) che integri tre strumenti di scansione di sicurezza standard del settore per fornire un'analisi completa della sicurezza del codice. Il server consente agli assistenti di codifica AI (come Kiro, Amazon Q Developer e Cline) di scansionare automaticamente frammenti di codice e configurazioni Infrastructure as Code (IaC). Con queste scansioni, gli assistenti di codifica possono aiutare a identificare vulnerabilità di sicurezza, configurazioni errate e violazioni della conformità.

I generatori di codice AI addestrati su milioni di frammenti di codice creano un punto cieco in materia di sicurezza: quanto erano sicuri i dati di addestramento? Questo modello fornisce una convalida della sicurezza in tempo reale durante la generazione del codice, aiutando gli sviluppatori a identificare e comprendere potenziali problemi di sicurezza durante la codifica. Questo approccio aiuta gli sviluppatori a risolvere sia le vulnerabilità dirette che i rischi ereditari derivanti dalle dipendenze. Colmando il divario tra efficienza dell'IA e conformità alla sicurezza, questo modello aiuta a consentire l'adozione sicura di strumenti di sviluppo basati sull'intelligenza artificiale.

Questo modello aiuta le organizzazioni a migliorare le proprie pratiche di sicurezza dello sviluppo attraverso strumenti di codifica assistiti dall'intelligenza artificiale, fornendo funzionalità di scansione di sicurezza continue su più linguaggi di programmazione e definizioni di infrastruttura. La soluzione combina le funzionalità dei seguenti strumenti:

- Checkov per la scansione di file IAC, inclusi Terraform e Kubernetes AWS CloudFormation manifest
- Semgrep per l'analisi di più linguaggi di programmazione come Python, JavaScript Java e altri
- Bandit per la scansione di sicurezza specializzata in Python

Le caratteristiche principali di questa soluzione includono quanto segue:

- Scansione delta di nuovi segmenti di codice, con riduzione del sovraccarico computazionale
- Ambienti isolati per strumenti di sicurezza, che impediscono la contaminazione tra strumenti
- Perfetta integrazione con gli assistenti di codifica AI (Kiro, Amazon Q Developer, Cline e altri)

- Feedback sulla sicurezza in tempo reale durante la generazione del codice
- Regole di scansione personalizzabili per la conformità organizzativa

Il modello fornisce un'interfaccia unificata per la scansione di sicurezza con formati di risposta standardizzati, che semplificano l'integrazione dei controlli di sicurezza nei flussi di lavoro di sviluppo. Il modello utilizza Python e il framework MCP per fornire feedback automatici sulla sicurezza. Questo approccio aiuta gli sviluppatori a identificare e risolvere i problemi di sicurezza nelle prime fasi del processo di sviluppo, apprendendo al contempo le migliori pratiche di sicurezza attraverso risultati dettagliati.

Prerequisiti e limitazioni

Prerequisiti

- Un programma attivo Account AWS con accesso all'utilizzo di Kiro o Amazon Q Developer, se desideri utilizzare uno di questi assistenti di codifica
- Python versione 3.10 o successiva installata
- uvgestore di pacchetti installato
- Familiarità con gli strumenti e i concetti di scansione di sicurezza
- Conoscenza di base di IaC e sicurezza delle applicazioni

Limitazioni

- La scansione Bandit è limitata ai soli file Python.
- La scansione in tempo reale potrebbe influire sulle prestazioni di basi di codice di grandi dimensioni.
- Le limitazioni specifiche degli strumenti si basano sui formati di file e sui linguaggi supportati.
- La revisione manuale è necessaria per convalidare i risultati di sicurezza.
- I risultati della scansione di sicurezza richiedono competenze in materia di sicurezza per una corretta interpretazione.
- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità regionale, vedi AWS Servizi per regione. Per endpoint specifici, consulta Service endpoints and quotas e scegli il link relativo al servizio.

Versioni del prodotto

- Python versione 3.10 o successiva
- Checkov versione 3.0.0 o successiva
- Semgrep versione 1.45.0 o successiva
- Bandit versione 1.7.5 o successiva
- MCP [cli] versione 1.11.0 o successiva
- Pydantic versione 1.10.0 o successiva
- Loguru versione 0.6.0 o successiva

Architecture

Il diagramma seguente mostra l'architettura di questa soluzione.

Il diagramma mostra il flusso di lavoro seguente:

1. Lo sviluppatore utilizza assistenti AI (ad esempio, Kiro, Cline, Amazon Q Developer o Roo Code) per generare o analizzare il codice. L'assistente AI invia il codice per la scansione di sicurezza.
2. Il server dello scanner di sicurezza MCP elabora la richiesta indirizzandola allo scanner specializzato appropriato: Checkov per i file IaC, Semgrep per l'analisi di più linguaggi di programmazione o Bandit per la scansione di sicurezza specifica di Python.
3. I risultati dello scanner con risultati di sicurezza, livelli di gravità, descrizioni dettagliate e correzioni suggerite vengono inviati allo sviluppatore tramite l'assistente AI.
4. Viene stabilito un ciclo di feedback continuo in cui lo sviluppatore riceve una convalida di sicurezza in tempo reale, abilita correzioni automatiche tramite assistenti AI e promuove le migliori pratiche di sicurezza durante lo sviluppo.

L'architettura mitiga i seguenti rischi comuni per la sicurezza:

- Iniezione di comandi
- Iniezione rapida
- Attraversamento del percorso
- Attacchi di dipendenza
- Esaurimento delle risorse

L'architettura mitiga questi rischi comuni per la sicurezza implementando le seguenti best practice:

- Tutti gli input dell'utente e del modello AI vengono scritti in file temporanei.
- Non vengono forniti input diretti ai comandi dell'interfaccia a riga di comando (CLI).
- L'accesso al file system è limitato solo alle directory e ai file temporanei.
- I file temporanei vengono puliti automaticamente.
- Le risposte di scansione vengono disinfettate.
- Viene applicato l'isolamento del processo che limita le capacità del processo.
- Tutte le attività di scansione vengono registrate.

Automazione e scalabilità

Il modello supporta l'automazione tramite le seguenti funzionalità:

- Integrazione con assistenti di codifica AI per la scansione automatica del codice
- Risposte API standardizzate per l'elaborazione automatizzata
- Configurazione tramite file di configurazione MCP
- Support per l'elaborazione in batch di più file
- Scansione scalabile su più linguaggi di programmazione e formati IAc

Il processo di scansione può essere automatizzato tramite gli endpoint API forniti:

- `scan_with_checkov`per la scansione IAC
- `scan_with_semgrep`per la scansione di codici multilingue
- `scan_with_bandit`per la scansione specifica di Python
- `get_supported_formats`per la convalida del formato

Quando estendete gli strumenti di scansione, seguite i principi di progettazione e le best practice descritti in precedenza in questa sezione. Vedi anche Best practice.

Tools (Strumenti)

Servizi AWS

- Kiro è un servizio di codifica agentica che collabora con gli sviluppatori per trasformare le istruzioni in specifiche dettagliate, quindi in codice funzionante, documenti e test. Gli agenti Kiro aiutano gli sviluppatori a risolvere problemi complessi e ad automatizzare attività come la generazione di documentazione e i test unitari.
- Amazon Q Developer è un assistente conversazionale generativo basato sull'intelligenza artificiale che può aiutarti a comprendere, creare, estendere e utilizzare le applicazioni. AWS

Altri strumenti

- Bandit è uno strumento di scansione di sicurezza Python specializzato. Rileva problemi di sicurezza comuni in Python come funzioni non sicure, segreti codificati e vulnerabilità di iniezione. Bandit fornisce valutazioni dettagliate di confidenza e gravità.
- Checkov è uno strumento statico di analisi del codice che verifica la presenza di configurazioni errate di sicurezza e conformità in IaC. Inoltre, Checkov rileva le violazioni della conformità e le migliori pratiche di sicurezza.
- Cline è un assistente di codifica basato sull'intelligenza artificiale che funziona in VS Code.
- Loguru è una libreria di convalida dei dati per Python.
- Model Context Protocol (MCP) è un framework open source per la creazione di strumenti di sviluppo assistiti dall'intelligenza artificiale.
- Pydantic è una libreria di convalida dei dati per Python.
- Semgrep analizza il codice sorgente alla ricerca di vulnerabilità e bug di sicurezza. Supporta più linguaggi di programmazione. Semgrep utilizza set di regole incentrati sulla sicurezza per un'analisi completa. Fornisce valutazioni dettagliate di confidenza e gravità.

Archivio di codice

Il codice di questo pattern è disponibile nell'archivio GitHub MCP Security Scanner: Real-Time Protection for AI Code Assistants. L'archivio include l'implementazione del server MCP, dettagli sulla configurazione MCP per Kiro, Amazon Q Developer, Cline e altri, esempi di configurazione e utilità di test.

La struttura del repository include:

- `security_scanner_mcp_server/`- Implementazione del server principale
- `docs/`- Documentazione e materiali dimostrativi

- `tests/`- File di test
- `mcp-config-example.json`- Esempio di configurazione MCP
- `requirements.txt`- Dipendenze del progetto

Best practice

Implementazione della scansione di

- Esamina i risultati di sicurezza per convalidare e dare priorità ai problemi.
- Mantieni gli strumenti di scansione (Checkov, Semgrep e Bandit) aggiornati alle versioni più recenti.
- Utilizzate lo strumento di sicurezza MCP di questo pattern insieme ad altre misure e strumenti di sicurezza.
- Aggiorna regolarmente i set di regole e le politiche di sicurezza.

Gestione della configurazione

- Utilizza i file di configurazione MCP nel codice sorgente ufficiale di controllo della versione.
- Documenta regole e configurazioni personalizzate.

Integration

- Integra la scansione di sicurezza nelle prime fasi del ciclo di sviluppo.
- Imposta la scansione automatica in hook pre-commit o in pipeline di integrazione continua e distribuzione continua (CI/CD).
- Configura le soglie di gravità appropriate per il tuo ambiente.
- Stabilisci procedure chiare per la gestione dei risultati di sicurezza.

Considerazioni operative

- Monitora le prestazioni di scansione e l'utilizzo delle risorse.
- Implementa la corretta gestione e registrazione degli errori.
- Conserva la documentazione delle configurazioni personalizzate.
- Stabilisci un processo per la revisione e l'aggiornamento delle regole di sicurezza.

Inoltre, tieni presente le seguenti best practice:

- Convalida sempre i risultati di sicurezza nel tuo contesto specifico.
- Mantieni aggiornati gli strumenti di sicurezza e le dipendenze.
- Utilizza più strumenti di sicurezza per una copertura completa.
- Segui le migliori pratiche di sicurezza nel tuo processo di sviluppo.

Epiche

(Utenti Kiro) Configura il server dello scanner di sicurezza MCP

Operazione	Description	Competenze richieste

Configurare le impostazioni MCP.	Puoi modificare i file di configurazione in Kiro (Opzione 1) localizzando manualmente i file di configurazione o (Opzione 2) utilizzando l'IDE Kiro.
	• (Opzione 1) Individua e modifica i file di configurazione manualmente come segue:
	  • Percorso: `.kiro/settings/mcp.json` (locale) o `~/.kiro/settings/mcp.json` (globale)
	• (Opzione 2) Modifica i file di configurazione utilizzando l'IDE Kiro come segue:
	  • Scegli il simbolo Kiro nel menu di navigazione a sinistra.
		Sviluppatore di app

Operazione	Description	Competenze richieste
	• Vai alla sezione «SERVER MCP». • Modifica le impostazioni di configurazione. Le modifiche apportate con l'IDE Kiro verranno salvate in uno di questi file di configurazione: `.kiro/settings/mcp.json` (locale) o `~/.kiro/settings/mcp.json` (globale) • (Entrambe le opzioni) Dopo aver completato l'opzione 1 o l'opzione 2, aggiungi la configurazione security-scanner `.kiro/settings/mcp.json` (locale) o `~/.kiro/settings/mcp.json` (globale): `{ "mcpServers": { "security-scanner" : { "command": "uvx", "args": [ "--from", "git+https://github.com/aws-samples/sample-mcp-security-scanner.git@main", "security_scanner_mcp_server"`	

Operazione	Description	Competenze richieste

	<pre>],
	      "env": {
	        "FASTMCP_
	LOG_LEVEL": "ERROR"
	      },
	      "disabled":
	 false,
	      "autoApprove": []
	    }
	  }
	}</pre>	

(Utenti Amazon Q Developer) Configura il server dello scanner di sicurezza MCP

Operazione	Description	Competenze richieste

Configurare le impostazioni MCP.	Per configurare manualmente le impostazioni MCP, effettuate le seguenti operazioni:
	1. Modificate il file di configurazione MCP in `~/.aws/amazonq/mcp.json` (globale) o `.amazonq/mcp.json` (locale).
	2. Aggiungete la seguente configurazione di security-scanner `~/.aws/amazonq/mcp.json` (globale) o `.amazonq/mcp.json` (locale):
	<pre>{
	  "mcpServers": {
	    "security-scanner"
	: {</pre>
		Sviluppatore di app

Operazione	Description	Competenze richieste

	<pre>"command": "uvx",
	"args": [
	"--from",
	"git+http
	s://github.com/aws
	-samples/sample-mc
	p-security-scanner
	.git@main",
	"security
	_scanner_mcp_server"
	],
	"env": {
	"FASTMCP_
	LOG_LEVEL": "ERROR"
	}
	}
	}
	}</pre>	

(Utenti Cline) Configurate il server dello scanner di sicurezza MCP

Operazione	Description	Competenze richieste
Configurare le impostazioni MCP.	Per configurare manualmente le impostazioni MCP, effettuate le seguenti operazioni: 1. Scegli l'estensione per aprirla, quindi seleziona Server MCP. 2. Scegli la scheda Installato, quindi scegli Configura server MCP per aprire il `cline_mcp_settings.json` file. 3. Aggiungi la seguente configurazione dello	Sviluppatore di app

Operazione	Description	Competenze richieste
	scanner di sicurezza: `cline_mcp_settings.json` `{ "mcpServers": { "security-scanner": { "command": "uvx", "args": [ "--from", "git+https://github.com/aws-samples/sample-mcp-security-scanner.git@main", "security_scanner_mcp_server" ], "env": { "FASTMCP_LOG_LEVEL": "ERROR" }, "disabled": false, "autoApprove": [] } } }`	

Esempio di analisi del codice con Python e Bandit

Operazione	Description	Competenze richieste
Esegui l'analisi del codice.	Per eseguire l'analisi del codice utilizzando Python e	Sviluppatore di app

Operazione	Description	Competenze richieste

	Bandit, utilizzate i seguenti passaggi:
	
	1. Crea un nuovo file Python denominato `bandit_test.py` con il seguente contenuto:
	
	`import pickle`
	`import yaml`
	`import subprocess`
	`from flask import Flask, request`
	`import hashlib`
	
	`app = Flask(__name__)`
	
	`@app.route("/unsafe_pickle", methods=["POST"])`
	`def unsafe_pickle_usage():`
	`    data = request.get_data()`
	`    return pickle.loads(data)`
	
	`@app.route("/command_injection", methods=["GET"])`
	`def command_injection():`
	`    command = request.args.get("cmd")`
	`    return subprocess.Popen(command, shell=True)`
	
	`def weak_crypto():`	

Operazione	Description	Competenze richieste

	`password = "secret_password" return hashlib.md5(password.encode()).hexdigest()`
	
	`PASSWORD = "super_secret_password123"`
	
	2. Apri l'assistente di codifica. Richiedi le seguenti azioni nella chat: Scansiona lo script corrente e dimmi i risultati.	

Esempio di analisi del codice utilizzando Terraform e Checkov

Operazione	Description	Competenze richieste
Esegui l'analisi del codice.	Per eseguire l'analisi del codice utilizzando Terraform e Checkov, utilizza i seguenti passaggi: 1. Crea un nuovo file Terraform denominato `terraform_test.tf` con il seguente contenuto: `resource "aws_s3_bucket" "insecure_bucket" { bucket = "my-insecure-bucket" acl = "public-read" }`	Sviluppatore di app

Operazione	Description	Competenze richieste

	``
	resource "aws_secu
	rity_group"
	 "wide_open" {
	  name =
	 "allow_all"
	  description =
	 "Allow all inbound
	 traffic"
	
	  ingress {
	  from_port =
	 0
	  to_port =
	 0
	  protocol =
	 "-1"
	  cidr_blocks =
	 ["0.0.0.0/0"]
	  }
	}
	`` 2. Apri l'assistente di codifica. Richiedi le seguenti azioni nella chat: Scansiona lo script corrente e dimmi i risultati.	

Esempio di funzionalità di scansione avanzate

Operazione	Description	Competenze richieste
Esegui scansioni mirate.	Di seguito sono riportati alcuni esempi di richieste che è possibile utilizzare	Sviluppatore di app

Operazione	Description	Competenze richieste
	per eseguire una scansione mirata: • Scansiona il progetto corrente e dimmi i risultati. • Scansiona le righe 6-10 dello script corrente.	
Usa la scansione di sicurezza con la generazione di codice.	Per risolvere i problemi di sicurezza utilizzando i loop di generazione del codice, utilizza i seguenti passaggi (questo esempio utilizza Kiro come assistente di codifica): 1. In Kiro, chiedi: crea una tabella DynamoDB in Terraform e scansiona il codice con lo scanner di sicurezza. 2. In Kiro, chiedi: esamina il codice generato e i risultati della scansione di sicurezza e segui i cicli di generazio ne del codice per risolvere i problemi di sicurezza.	Sviluppatore di app

Risoluzione dei problemi

Problema	Soluzione
Problemi di configurazione dell'ambiente	• Verifica che sia installata la versione 3.10 o successiva di Python.

Problema	Soluzione
	• Assicurati che il gestore di uv pacchetti sia installato.
Problemi con lo scanner	• Verificare che i formati di file siano supportati. • Verifica che la sintassi del file sia valida. • Assicurati che vengano utilizzate le estension i di file corrette. • Rivedi le impostazioni del timeout di scansione.
Problemi di integrazione	• Verificare che il server MCP sia in esecuzion e. • Verificate che il file di configurazione sia corretto. • Convalida gli endpoint dell'API. • Assicurati che il formato di risposta sia valido.
Supporto aggiuntivo	• Esamina i log per i messaggi di errore dettagliati. • Consulta la documentazione specifica dello strumento. • Crea un problema nel repository.

Risorse correlate

AWS documentazione

- Infrastruttura come codice (AWS Whitepaper Introduzione a on) DevOps AWS

Altre risorse AWS

- Best practice per sicurezza, identità e conformità.

Altre risorse

- Documentazione Bandit
- Documentazione Checkov
- Documentazione del Model Context Protocol (MCP)
- OWASP Secure Coding Practices (sito web della OWASP Foundation)
- Documentazione Semgrep

Informazioni aggiuntive

Esempio di configurazione MCP con approvazione automatica abilitata

Se non è `autoApprove` configurato, l'utente deve concedere l'approvazione per inviare il codice al server di sicurezza MCP per la scansione. Una volta `autoApprove` configurato, l'assistente al codice può richiamare gli strumenti senza l'approvazione dell'utente. Questi strumenti vengono eseguiti localmente sulla macchina, non viene inviato alcun dato e viene eseguita solo una scansione del codice.

La seguente configurazione consente l'esecuzione automatica di tutte le funzioni di scansione di sicurezza:

```
{
  "mcpServers": {
    "security-scanner": {
      "command": "uvx",
      "args": [
        "--from",
        "git+https://github.com/aws-samples/sample-mcp-security-scanner.git@main",
        "security_scanner_mcp_server"
      ],
      "env": {
        "FASTMCP_LOG_LEVEL": "ERROR"
      },
      "disabled": false,
      "autoApprove": [
        "scan_with_checkov",
        "scan_with_semgrep",
        "scan_with_bandit",
        "get_supported_formats"
      ]
    }
  }
```

```
}
```

Per abilitare la registrazione di debug, impostare su`"DEBUG"`. `"FASTMCP_LOG_LEVEL"`

Formati di file supportati dagli strumenti di scansione di sicurezza

Ogni strumento di scansione di sicurezza di questa soluzione supporta i seguenti formati di file:

Checkov (IaC)

- Terraform: .tf, .tfvars, .tfstate
- CloudFormation — .yaml, .yml, .json, .template
- Kubernetes: .yaml, .yml
- Dockerfile — Dockerfile
- ARM — .json (Gestore delle risorse di Azure)
- Bicipite — .bicipite
- Senza server: .yml, .yaml
- Timone: .yaml, .yml, .tpl
- GitHub Azioni: .yml, .yaml
- GitLab_ci — .yml, .yaml
- Ansible: .yml, .yaml

Semgrep (codice sorgente)

- Python — .py
- JavaScript — .js
- TypeScript — .ts
- Giava — .java
- Vai — .go
- C — c.
- C++ — .cpp
- C# — .cs
- Rubino — .rb
- PHP — .php

- Scala — .scala
- Kotlin — .kt
- Rust — .rs

Bandit (solo Python)

- Python — .py

Demo

Per la scansione del codice, prova i seguenti esempi di istruzioni con il tuo assistente AI:

- «Scansiona lo script corrente e dimmi i risultati».
- «Scansiona le righe da 20 a 60 e dimmi i risultati».
- «Scansiona questa risorsa della tabella Amazon DynamoDB e dimmi il risultato».

Per ulteriori informazioni, consulta questa demo sulla scansione del codice nell'archivio di GitHub questo pattern.

Per generare codice sicuro, prova i seguenti prompt di esempio:

- «Genera una configurazione Terraform per creare una tabella DynamoDB con crittografia abilitata e scansionala per problemi di sicurezza».
- «Crea una funzione Python Lambda che scriva su DynamoDB e scansionala per individuare eventuali vulnerabilità».
- «Genera un CloudFormation modello per un bucket S3 con impostazioni di sicurezza adeguate e verifica che superi i controlli di sicurezza».
- «Scrivi uno script Python per interrogare DynamoDB con impaginazione e scansiona le migliori pratiche di sicurezza».
- «Crea un manifesto di distribuzione Kubernetes per un microservizio con un rafforzamento della sicurezza e convalidalo».

Per ulteriori informazioni, guarda questa demo sulla generazione di codice con scansione di sicurezza nell'archivio di questo pattern. GitHub

Sviluppa assistenti avanzati basati sull'intelligenza artificiale generativa utilizzando RAG e suggerimenti ReAct

Praveen Kumar Jeyarajan, Shuai Cao, Noah Hamilton, Kiowa Jackson, Jundong Qiao e Kara Yang, Amazon Web Services

Riepilogo

Un'azienda tipica ha il 70 percento dei propri dati intrappolati in sistemi isolati. Puoi utilizzare assistenti generativi basati su chat basati sull'intelligenza artificiale per sbloccare informazioni e relazioni tra questi silos di dati attraverso interazioni in linguaggio naturale. Per ottenere il massimo dall'intelligenza artificiale generativa, i risultati devono essere affidabili, accurati e includere i dati aziendali disponibili. Il successo degli assistenti basati sulla chat dipende da quanto segue:

- Modelli di intelligenza artificiale generativa (come Anthropic Claude 2)
- Vettorizzazione delle fonti di dati
- Tecniche di ragionamento avanzate, come il ReAct framework, per suggerire il modello

Questo modello fornisce approcci per il recupero dei dati da fonti di dati come bucket Amazon Simple Storage Service (Amazon S3), AWS Glue e Amazon Relational Database Service (Amazon RDS). Si ottiene valore da tali dati combinando Retrieval Augmented Generation (RAG) con i metodi. chain-of-thought I risultati supportano complesse conversazioni con assistenti basate su chat che attingono alla totalità dei dati archiviati dall'azienda.

Questo modello utilizza i SageMaker manuali di Amazon e le tabelle dei dati dei prezzi come esempio per esplorare le funzionalità di un assistente generativo basato sull'intelligenza artificiale basato su chat. Creerai un assistente basato su chat che aiuterà i clienti a valutare il SageMaker servizio rispondendo a domande sui prezzi e sulle funzionalità del servizio. La soluzione utilizza una libreria Streamlit per la creazione dell'applicazione frontend e il LangChain framework per lo sviluppo del backend dell'applicazione basato su un modello di linguaggio di grandi dimensioni (LLM).

Le richieste all'assistente basato sulla chat vengono soddisfatte con una classificazione iniziale degli intenti per il routing verso uno dei tre possibili flussi di lavoro. Il flusso di lavoro più sofisticato combina una consulenza generale con un'analisi complessa dei prezzi. È possibile adattare il modello ai casi d'uso aziendali, aziendali e industriali.

Prerequisiti e limitazioni

Prerequisiti

- AWS Command Line Interface (AWS CLI) installata e configurata
- AWS Cloud Development Kit (AWS CDK) Toolkit 2.114.1 o versione successiva installato e configurato
- Familiarità di base con Python e AWS CDK
- Git installato
- Docker installato
- Python 3.11 o successivo installato e configurato (per maggiori informazioni, consulta la sezione Strumenti)
- Un account AWS attivo avviato utilizzando AWS CDK
- Accesso ai modelli Amazon Titan e Anthropic Claude abilitato nel servizio Amazon Bedrock
- Credenziali di sicurezza AWS`AWS_ACCESS_KEY_ID`, incluse quelle configurate correttamente nell'ambiente terminale

Limitazioni

- LangChain non supporta tutti i LLM per lo streaming. I modelli Anthropic Claude sono supportati, ma i modelli di AI21 Labs no.
- Questa soluzione viene distribuita su un singolo account AWS.
- Questa soluzione può essere distribuita solo nelle regioni AWS in cui sono disponibili Amazon Bedrock e Amazon Kendra. Per informazioni sulla disponibilità, consulta la documentazione per Amazon Bedrock e Amazon Kendra.

Versioni del prodotto

- Python versione 3.11 o successiva
- Streamlit versione 1.30.0 o successiva
- Streamlit-chat versione 0.1.1 o successiva
- LangChain versione 0.1.12 o successiva
- AWS CDK versione 2.132.1 o successiva

Architecture

Stack tecnologico Target

- Amazon Athena
- Amazon Bedrock
- Amazon Elastic Container Service (Amazon ECS)
- AWS Glue
- AWS Lambda
- Simple Storage Service (Amazon S3)
- Amazon Kendra
- Elastic Load Balancing

Architettura di destinazione

Il codice CDK AWS distribuirà tutte le risorse necessarie per configurare l'applicazione di assistenza basata su chat in un account AWS. L'applicazione di assistenza basata sulla chat mostrata nel diagramma seguente è progettata per rispondere alle domande correlate degli utenti. SageMaker Gli utenti si connettono tramite un Application Load Balancer a un VPC che contiene un cluster Amazon ECS che ospita l'applicazione Streamlit. Una funzione Lambda di orchestrazione si connette all'applicazione. Le sorgenti dati dei bucket S3 forniscono dati alla funzione Lambda tramite Amazon Kendra e AWS Glue. La funzione Lambda si connette ad Amazon Bedrock per rispondere alle domande (domande) degli utenti assistenti basati sulla chat.

1. La funzione di orchestrazione Lambda invia la richiesta di prompt LLM al modello Amazon Bedrock (Claude 2).
2. Amazon Bedrock invia la risposta LLM alla funzione di orchestrazione Lambda.

Flusso logico all'interno della funzione Lambda di orchestrazione

Quando gli utenti fanno una domanda tramite l'applicazione Streamlit, richiama direttamente la funzione di orchestrazione Lambda. Il diagramma seguente mostra il flusso logico quando viene richiamata la funzione Lambda.

- Fase 1 — L'input `query` (domanda) viene classificato in uno dei tre intenti:
  - Domande generali di SageMaker orientamento
  - Domande generali SageMaker sui prezzi (formazione/inferenza)
  - Domande complesse relative ai prezzi SageMaker
- Fase 2 — L'input `query` avvia uno dei tre servizi:
  - `RAG Retrieval service`, che recupera il contesto pertinente dal database vettoriale Amazon Kendra e richiama l'LLM tramite Amazon Bedrock per riepilogare il contesto recuperato come risposta.
  - `Database Query service`, che utilizza il LLM, i metadati del database e le righe di esempio delle tabelle pertinenti per convertire l'input in una query SQL. `query` Il servizio Database Query esegue la query SQL sul database SageMaker dei prezzi tramite Amazon Athena e riassume i risultati della query come risposta.
  - `In-context ReACT Agent service`, che suddivide l'input `query` in più fasi prima di fornire una risposta. L'agente utilizza `RAG Retrieval service` e `Database Query service` come strumenti per recuperare le informazioni pertinenti durante il processo di ragionamento. Una volta completati i processi di ragionamento e azione, l'agente genera la risposta finale come risposta.
- Fase 3 — La risposta della funzione Lambda di orchestrazione viene inviata all'applicazione Streamlit come output.

Tools (Strumenti)

Servizi AWS

- Amazon Athena è un servizio di query interattivo che ti aiuta ad analizzare i dati direttamente in Amazon Simple Storage Service (Amazon S3) utilizzando SQL standard.
- Amazon Bedrock è un servizio completamente gestito che rende disponibili per l'uso modelli di base ad alte prestazioni (FMs) delle principali startup di intelligenza artificiale e di Amazon tramite un'API unificata.
- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software che aiuta a definire e fornire l'infrastruttura cloud AWS in codice.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.

- Amazon Elastic Container Service (Amazon ECS) è un servizio rapido e scalabile di gestione dei container che ti aiuta a eseguire, arrestare e gestire container in un cluster.
- AWS Glue è un servizio di estrazione, trasformazione e caricamento (ETL) completamente gestito. Ti aiuta a classificare, pulire, arricchire e spostare i dati in modo affidabile tra archivi di dati e flussi di dati. Questo modello utilizza un crawler AWS Glue e una tabella AWS Glue Data Catalog.
- Amazon Kendra è un servizio di ricerca intelligente che utilizza l'elaborazione del linguaggio naturale e algoritmi avanzati di apprendimento automatico per restituire risposte specifiche alle domande di ricerca dai tuoi dati.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Elastic Load Balancing (ELB) distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni. Ad esempio, puoi distribuire il traffico tra istanze Amazon Elastic Compute Cloud (Amazon EC2), contenitori e indirizzi IP in una o più zone di disponibilità.

Repository di codice

Il codice per questo pattern è disponibile nel GitHub genai-bedrock-chatbotrepository.

L'archivio del codice contiene i seguenti file e cartelle:

- `assets`cartella: le risorse statiche, il diagramma dell'architettura e il set di dati pubblico
- `code/lambda-container`folder — Il codice Python eseguito nella funzione Lambda
- `code/streamlit-app`folder — Il codice Python che viene eseguito come immagine del contenitore in Amazon ECS
- `tests`folder — I file Python che vengono eseguiti per testare unitariamente i costrutti CDK di AWS
- `code/code_stack.py`— Il CDK AWS costruisce i file Python utilizzati per creare risorse AWS
- `app.py`— I file Python dello stack CDK di AWS utilizzati per distribuire le risorse AWS nell'account AWS di destinazione
- `requirements.txt`— L'elenco di tutte le dipendenze Python che devono essere installate per AWS CDK
- `requirements-dev.txt`— L'elenco di tutte le dipendenze Python che devono essere installate affinché AWS CDK esegua la suite unit-test

- `cdk.json`— Il file di input per fornire i valori necessari per avviare le risorse

Nota: il codice CDK di AWS utilizza costrutti L3 (livello 3) e policy AWS Identity and Access Management (IAM) gestite da AWS per distribuire la soluzione.

Best practice

- L'esempio di codice fornito qui è solo per una demo proof-of-concept (PoC) o pilota. Se vuoi portare il codice in Production, assicurati di utilizzare le seguenti best practice:
  - La registrazione degli accessi di Amazon S3 è abilitata.
  - I log di flusso VPC sono abilitati.
  - L'indice Amazon Kendra Enterprise Edition è abilitato.
- Imposta il monitoraggio e gli avvisi per la funzione Lambda. Per ulteriori informazioni, consulta Monitoraggio e risoluzione dei problemi delle funzioni Lambda. Per le best practice generali relative all'utilizzo delle funzioni Lambda, consulta la documentazione AWS.

Epiche

Configura le credenziali AWS sul tuo computer locale

Operazione	Description	Competenze richieste

Esporta le variabili per l'account e la regione AWS in cui verrà distribuito lo stack.	Per fornire le credenziali AWS per AWS CDK utilizzando variabili di ambiente, esegui i seguenti comandi.
	`export CDK_DEFAU LT_ACCOUNT=<12 Digit AWS Account Number>`
	`export CDK_DEFAU LT_REGION=<region>`
		DevOps ingegnere, AWS DevOps

Configura il profilo AWS CLI.	Per configurare il profilo AWS CLI per l'account, segui le	DevOps ingegnere, AWS DevOps

Operazione	Description	Competenze richieste
	istruzioni nella documenta zione AWS.	

Configurare l'ambiente

Operazione	Description	Competenze richieste
Clona il repository sul tuo computer locale.	Per clonare il repository, esegui il seguente comando nel tuo terminale. `git clone https://github.com/awslabs/genai-bedrock-chatbot.git`	DevOps ingegnere, AWS DevOps

Configura l'ambiente virtuale Python e installa le dipendenze richieste.	Per configurare l'ambiente virtuale Python, esegui i seguenti comandi.
	`cd genai-bedrock-chatbot`
	`python3 -m venv .venv`
	`source .venv/bin/activate`
	Per configurare le dipendenze richieste, esegui il comando seguente.
	`pip3 install -r requirements.txt`
		DevOps ingegnere, AWS DevOps

Configura l'ambiente AWS CDK e sintetizza il codice CDK AWS.	1. Per configurare l'ambiente AWS CDK nel tuo account	DevOps ingegnere, AWS DevOps

Operazione	Description	Competenze richieste
	AWS, esegui il comando seguente. `cdk bootstrap aws://ACCOUNT-NUMBER/REGION` 2. Per convertire il codice in una configurazione CloudFormation dello stack AWS, esegui il comando`cdk synth`.	

Configura e distribuisci l'applicazione di assistenza basata sulla chat

Operazione	Description	Competenze richieste
Fornire l'accesso al modello Claude.	Per abilitare l'accesso al modello Anthropic Claude per il tuo account AWS, segui le istruzioni nella documenta zione di Amazon Bedrock.	AWS DevOps
Distribuisci risorse nell'acco unt.	Per distribuire risorse nell'acco unt AWS utilizzando il CDK AWS, procedi come segue: 1. Nella radice del repositor y clonato, nel `cdk.json` file, fornisci gli input per i parametri. `logging` I valori di esempio sono`INFO`,, e`DEBUG`. `WARN` `ERROR` Questi valori definisco no i messaggi a livello di	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	registro per la funzione Lambda e l'applicazione Streamlit. 2. Il `app.py` file nella radice del repository clonato contiene il nome dello CloudFormation stack AWS utilizzato per la distribuz ione. Il nome dello stack predefinito è. `chatbot-s tack` 3. Per distribuire risorse, esegui il comando. `cdk deploy` Il `cdk deploy` comando utilizza costrutti L3 per creare più funzioni Lambda per copiare documenti e file di set di dati CSV nei bucket S3. 4. Una volta completato il comando, accedi alla Console di gestione AWS, apri la CloudFormation console e verifica che lo stack sia stato distribuito correttamente. Una volta completata la distribuzione, puoi accedere all'applicazione di assistenz a basata sulla chat utilizzan	

Operazione	Description	Competenze richieste
	do l'URL fornito nella sezione Output. CloudFormation	

Operazione	Description	Competenze richieste
Esegui il crawler AWS Glue e crea la tabella Data Catalog.	Un crawler AWS Glue viene utilizzato per mantenere dinamico lo schema dei dati. La soluzione crea e aggiorna le partizioni nella tabella del catalogo dati di AWS Glue eseguendo il crawler su richiesta. Dopo aver copiato i file del set di dati CSV nel bucket S3, esegui il crawler AWS Glue e crea lo schema della tabella Data Catalog per i test: 1. Accedi alla console AWS Glue. 2. Nel pannello di navigazione, in Data Catalog, scegli Crawlers. 3. Seleziona il crawler con il suffisso. `sagemaker-pricing-crawler` 4. Esegui il crawler. 5. Una volta eseguito correttamente, il crawler crea una tabella AWS Glue Data Catalog. **Note** Il codice CDK AWS configura il crawler AWS Glue per l'esecuzione su	DevOps ingegnere, AWS DevOps

Operazione	Description	Competenze richieste
	richiesta, ma puoi anche pianificarne l'esecuzione periodica.	

Operazione	Description	Competenze richieste
Avvia l'indicizzazione dei documenti.	Dopo aver copiato i file nel bucket S3, usa Amazon Kendra per scansionarli e indicizzarli: 1. Accedi alla console Amazon Kendra. 2. Seleziona l'indice con il suffisso. `chatbot-index` 3. Nel riquadro di navigazione, scegli Origini dati e seleziona il connettore di origine dati con il suffisso`chatbot-index` . 4. Scegli Sincronizza ora per avviare il processo di indicizzazione. **Note** Il codice CDK AWS configura la sincronizzazione dell'indice Amazon Kendra per l'esecuzione su richiesta, ma puoi anche eseguirla periodicamente utilizzando il parametro Schedule.	AWS DevOps, DevOps ingegnere

Pulisci tutte le risorse AWS nella soluzione

Operazione	Description	Competenze richieste
Rimuovi le risorse AWS.	Dopo aver testato la soluzione , pulisci le risorse: 1. Per rimuovere le risorse AWS distribuite dalla soluzione, esegui il comando`cdk destroy`. 2. Elimina tutti gli oggetti dai due bucket S3, quindi rimuovi i bucket. Per ulteriori informazioni, consulta Eliminazione di un bucket.	DevOps ingegnere, AWS DevOps

Risoluzione dei problemi

Problema	Soluzione
AWS CDK restituisce errori.	Per assistenza con i problemi di AWS CDK, consulta Risoluzione dei problemi comuni di AWS CDK.

Risorse correlate

- Amazon Base:
  - Accesso al modello
  - Parametri di inferenza per i modelli di base
- Creazione di funzioni Lambda con Python
- Inizia a usare il CDK AWS
- Lavorare con il CDK AWS in Python

- Generative AI Application Builder su AWS
- LangChain documentazione
- Documentazione semplificata

Informazioni aggiuntive

Comandi AWS CDK

Quando lavori con AWS CDK, tieni presente i seguenti comandi utili:

- Elenca tutti gli stack presenti nell'app

```
cdk ls
```

- Emette il modello AWS sintetizzato CloudFormation

```
cdk synth
```

- Distribuisce lo stack nell'account e nella regione AWS predefiniti

```
cdk deploy
```

- Confronta lo stack distribuito con lo stato attuale

```
cdk diff
```

- Apre la documentazione di AWS CDK

```
cdk docs
```

- Elimina lo CloudFormation stack e rimuove le risorse distribuite da AWS

```
cdk destroy
```

Sviluppa un assistente basato su chat completamente automatizzato utilizzando gli agenti e le knowledge base di Amazon Bedrock

Jundong Qiao, Shuai Cao, Noah Hamilton, Kiowa Jackson, Praveen Kumar Jeyarajan e Kara Yang, Amazon Web Services

Riepilogo

Molte organizzazioni incontrano difficoltà nella creazione di un assistente basato su chat in grado di orchestrare diverse fonti di dati per offrire risposte complete. Questo modello presenta una soluzione per lo sviluppo di un assistente basato su chat in grado di rispondere alle domande provenienti sia dalla documentazione che dai database, con un'implementazione semplice.

A partire da Amazon Bedrock, questo servizio di intelligenza artificiale generativa (AI) completamente gestito offre un'ampia gamma di modelli di base avanzati (FMs). Ciò facilita la creazione efficiente di applicazioni di intelligenza artificiale generativa con una forte attenzione alla privacy e alla sicurezza. Nel contesto del recupero della documentazione, il Retrieval Augmented Generation (RAG) è una funzionalità fondamentale. Utilizza basi di conoscenza per arricchire i prompt FM con informazioni contestualmente rilevanti provenienti da fonti esterne. Un indice Amazon OpenSearch Serverless funge da database vettoriale alla base delle knowledge base per Amazon Bedrock. Questa integrazione è migliorata attraverso un'attenta progettazione tempestiva per ridurre al minimo le imprecisioni e garantire che le risposte siano ancorate a una documentazione fattuale. Per le query sui database, Amazon Bedrock trasforma le FMs richieste testuali in query SQL strutturate, incorporando parametri specifici. Ciò consente il recupero preciso dei dati dai database gestiti dai database.AWS Glue Amazon Athena viene utilizzato per queste query.

Per gestire interrogazioni più complesse, ottenere risposte complete richiede informazioni provenienti sia dalla documentazione che dai database. Agents for Amazon Bedrock è una funzionalità di intelligenza artificiale generativa che ti aiuta a creare agenti autonomi in grado di comprendere attività complesse e suddividerle in attività più semplici da orchestrare. La combinazione di informazioni ricavate dalle attività semplificate, facilitata dagli agenti autonomi di Amazon Bedrock, migliora la sintesi delle informazioni, portando a risposte più complete ed esaustive. Questo modello dimostra come creare un assistente basato su chat utilizzando Amazon Bedrock e i relativi servizi e funzionalità di intelligenza artificiale generativa all'interno di una soluzione automatizzata.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Docker, installato
- AWS Cloud Development Kit (AWS CDK), installato e avviato in o `us-east-1` `us-west-2` Regioni AWS
- AWS CDK Toolkit versione 2.114.1 o successiva, installata
- AWS Command Line Interface (AWS CLI), installato e configurato
- Python versione 3.11 o successiva, installata
- In Amazon Bedrock, abilita l'accesso a Claude 2, Claude 2.1, Claude Instant e Titan Embeddings G1 — Text

Limitazioni

- Questa soluzione viene distribuita su un unico dispositivo. Account AWS
- Questa soluzione può essere implementata solo Regioni AWS dove sono supportati Amazon Bedrock e Amazon OpenSearch Serverless. Per ulteriori informazioni, consulta la documentazione per Amazon Bedrock e Amazon OpenSearch Serverless.

Versioni del prodotto

- Llama-index versione 0.10.6 o successiva
- Sqlalchemy versione 2.0.23 o successiva
- OpenSearch-PY versione 2.4.2 o successiva
- Requests_AWS4Auth versione 1.2.3 o successiva
- AWS SDK per Python (Boto3) versione 1.34.57 o successiva

Architecture

Stack tecnologico Target

AWS Cloud Development Kit (AWS CDK)È un framework di sviluppo software open source per definire l'infrastruttura cloud in codice e fornirla tramite. AWS CloudFormation Lo AWS CDK stack utilizzato in questo modello distribuisce le seguenti risorse: AWS

- AWS Key Management Service (AWS KMS)
- Amazon Simple Storage Service (Amazon S3)
- AWS Glue Data Catalog, per il componente del database AWS Glue
- AWS Lambda
- AWS Identity and Access Management (IAM)
- Amazon OpenSearch Serverless
- Amazon Elastic Container Registry (Amazon ECR)
- Amazon Elastic Container Service (Amazon ECS)
- AWS Fargate
- Amazon Virtual Private Cloud (Amazon VPC) (Amazon VPC)
- Application Load Balancer

Architettura Target

Il diagramma mostra una configurazione AWS nativa del cloud completa all'interno di un'unica configurazione, utilizzandone più di una. Regione AWS Servizi AWS L'interfaccia principale per l'assistente basato sulla chat è un'applicazione Streamlit ospitata su un cluster Amazon ECS. Un Application Load Balancer gestisce l'accessibilità. Le interrogazioni effettuate tramite questa interfaccia attivano la funzione `Invocation` Lambda, che si interfaccia quindi con gli agenti per Amazon Bedrock. Questo agente risponde alle richieste degli utenti consultando le knowledge base per Amazon Bedrock o richiamando una funzione Lambda. `Agent executor` Questa funzione attiva una serie di azioni associate all'agente, seguendo uno schema API predefinito. Le knowledge base di Amazon Bedrock utilizzano un indice OpenSearch Serverless come base per il database vettoriale. Inoltre, la `Agent executor` funzione genera query SQL che vengono eseguite sul AWS Glue database tramite Amazon Athena.

Tools (Strumenti)

Servizi AWS

- Amazon Athena è un servizio di query interattivo che ti aiuta ad analizzare i dati direttamente in Amazon Simple Storage Service (Amazon S3) utilizzando SQL standard.
- Amazon Bedrock è un servizio completamente gestito che rende disponibili per l'uso modelli di base ad alte prestazioni (FMs) delle principali startup di intelligenza artificiale e di Amazon tramite un'API unificata.
- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che ti aiuta a definire e fornire l'infrastruttura AWS Cloud in codice.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando.
- Amazon Elastic Container Service (Amazon ECS) è un servizio rapido e scalabile di gestione dei container che ti aiuta a eseguire, arrestare e gestire container in un cluster.
- Elastic Load Balancing distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni. Ad esempio, puoi distribuire il traffico tra istanze Amazon Elastic Compute Cloud (Amazon EC2), contenitori e indirizzi IP in una o più zone di disponibilità.
- AWS Glueè un servizio di estrazione, trasformazione e caricamento (ETL) completamente gestito. Ti aiuta a classificare, pulire, arricchire e spostare i dati in modo affidabile tra archivi di dati e flussi di dati. Questo modello utilizza un AWS Glue crawler e una tabella. AWS Glue Data Catalog
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon OpenSearch Serverless è una configurazione serverless su richiesta per Amazon Service. OpenSearch In questo modello, un indice OpenSearch Serverless funge da database vettoriale per le knowledge base di Amazon Bedrock.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri strumenti

- Streamlit è un framework Python open source per creare applicazioni di dati.

Deposito di codice

Il codice per questo pattern è disponibile nel GitHub genai-bedrock-agent-chatbotrepository. L'archivio del codice contiene i seguenti file e cartelle:

- `assets`cartella: le risorse statiche, come il diagramma dell'architettura e il set di dati pubblico.
- `code/lambdas/action-lambda`folder — Il codice Python per la funzione Lambda che funge da azione per l'agente Amazon Bedrock.
- `code/lambdas/create-index-lambda`folder — Il codice Python per la funzione Lambda che crea l'indice Serverless. OpenSearch
- `code/lambdas/invoke-lambda`folder — Il codice Python per la funzione Lambda che richiama l'agente Amazon Bedrock, chiamato direttamente dall'applicazione Streamlit.
- `code/lambdas/update-lambda`folder — Il codice Python per la funzione Lambda che aggiorna o elimina le risorse dopo che le risorse sono state distribuite tramite. AWS AWS CDK
- `code/layers/boto3_layer`folder: lo AWS CDK stack che crea un livello Boto3 condiviso tra tutte le funzioni Lambda.
- `code/layers/opensearch_layer`folder: lo AWS CDK stack che crea un livello OpenSearch Serverless che installa tutte le dipendenze per creare l'indice.
- `code/streamlit-app`folder — Il codice Python che viene eseguito come immagine del contenitore in Amazon ECS.
- `code/code_stack.py`— I file di AWS CDK costruzione in Python che AWS creano risorse.
- `app.py`— I file Python AWS CDK dello stack che AWS distribuiscono le risorse nell'account di destinazione. AWS
- `requirements.txt`— L'elenco di tutte le dipendenze Python che devono essere installate per. AWS CDK
- `cdk.json`— Il file di input per fornire i valori necessari per creare risorse. Inoltre, nei `context/config` campi, è possibile personalizzare la soluzione di conseguenza. Per ulteriori informazioni sulla personalizzazione, vedere la sezione Informazioni aggiuntive.

Best practice

- L'esempio di codice fornito qui è solo a scopo proof-of-concept (PoC) o pilota. Se vuoi portare il codice in produzione, assicurati di utilizzare le seguenti best practice:
  - Abilita la registrazione degli accessi ad Amazon S3
  - Abilita i log di flusso VPC
- Imposta il monitoraggio e gli avvisi per le funzioni Lambda. Per ulteriori informazioni, consulta Monitoraggio e risoluzione dei problemi delle funzioni Lambda. Per le best practice, consulta le Best practice per l'utilizzo AWS Lambda delle funzioni.

Epiche

Imposta AWS le credenziali sulla tua workstation locale

Operazione	Description	Competenze richieste

Esporta le variabili per l'account e la regione.	Per fornire AWS le credenziali AWS CDK utilizzando le variabili di ambiente, esegui i seguenti comandi.
	`export CDK_DEFAULT_ACCOUNT=<12-digit AWS account number>`
	`export CDK_DEFAULT_REGION=<Region>`
		AWS DevOps, DevOps ingegnere

Configura il profilo AWS CLI denominato.	Per configurare il profilo AWS CLI denominato per l'account, segui le istruzioni in Configurazione e impostazioni del file di credenziali.	AWS DevOps, DevOps ingegnere

Configurare l'ambiente

Operazione	Description	Competenze richieste
Clona il repository sulla tua workstation locale.	Per clonare il repository, esegui il seguente comando nel tuo terminale. `git clone https://github.com/awslabs/genai-bedrock-agent-chatbot.git`	DevOps ingegnere, AWS DevOps

Operazione	Description	Competenze richieste

Configura l'ambiente virtuale Python.	Per configurare l'ambient e virtuale Python, esegui i seguenti comandi.
	`cd genai-bedrock-agent-chatbot`
	`python3 -m venv .venv`
	`source .venv/bin/activate`
	Per configurare le dipendenz e richieste, esegui il comando seguente.
	`pip3 install -r requirements.txt`
		DevOps ingegnere, AWS DevOps

Configura l' AWS CDK ambiente.	Per convertire il codice in un AWS CloudForm ation modello, esegui il comando`cdk synth`.	AWS DevOps, DevOps ingegnere

Configura e distribuisci l'applicazione

Operazione	Description	Competenze richieste
Distribuisci risorse nell'acco unt.	Per distribuire risorse Account AWS utilizzando il AWS CDK, procedi come segue: 1. Nella radice del repositor y clonato, nel `cdk.json` file, fornite gli input per i parametri di registraz ione. I valori di esempio	DevOps ingegnere, AWS DevOps

Operazione	Description	Competenze richieste
	sonoINFO,, e. DEBUG WARN ERROR Questi valori definisco no i messaggi a livello di registro per le funzioni Lambda e l'applicazione Streamlit. 2. Il cdk.json file nella radice del repository clonato contiene il nome dello stack utilizzato per la distribuz ione. AWS CloudFormation Il nome dello stack predefini to è. chatbot-stack Il nome dell'agente Amazon Bedrock predefinito è e ChatbotBedrockAgen t l'alias dell'agente Amazon Bedrock predefini to è. Chatbot_Agent 3. Per distribuire risorse, esegui il comando. cdk deploy Il cdk deploy comando utilizza costrutti di livello 3 per creare più funzioni Lambda per copiare documenti e file di set di dati CSV nei bucket S3. Implementa inoltre l'agente Amazon Bedrock, le knowledge base e la funzione Action group Lambda per l'agente Amazon Bedrock.	

Operazione	Description	Competenze richieste
	4. Accedi a, quindi Console di gestione AWS apri la console.CloudFormation 5. Verifica che lo stack sia stato distribuito correttamente. Per istruzioni, vedi Revisione dello stack sulla console. AWS CloudFormation Una volta completata la distribuzione, puoi accedere all'applicazione di assistenza basata sulla chat utilizzando l'URL fornito nella scheda Output della console. CloudFormation	

Pulisci tutte AWS le risorse della soluzione

Operazione	Description	Competenze richieste
Rimuovi le AWS risorse.	Dopo aver testato la soluzione, per ripulire le risorse, esegui il comando `cdk destroy`.	AWS DevOps, DevOps ingegnere

Risorse correlate

AWS documentazione

- Risorse Amazon Bedrock:
  - Accesso al modello
  - Parametri di inferenza per i modelli di base

  - Agenti per Amazon Bedrock
  - Basi di conoscenza per Amazon Bedrock
- Creazione di funzioni Lambda con Python
- AWS CDK risorse:
  - Inizia con AWS CDK
  - Risoluzione dei AWS CDK problemi più comuni
  - Lavorare con il AWS CDK in Python
- Generative AI Application Builder attivo AWS

Altre risorse AWS

- Motore vettoriale per Amazon Serverless OpenSearch

Altre risorse

- LlamaIndex documentazione
- Documentazione semplificata

Informazioni aggiuntive

Personalizza l'assistente basato sulla chat con i tuoi dati

Per integrare i dati personalizzati per l'implementazione della soluzione, segui queste linee guida strutturate. Questi passaggi sono progettati per garantire un processo di integrazione semplice ed efficiente, che ti consenta di implementare la soluzione in modo efficace con i tuoi dati personalizzati.

Per l'integrazione dei dati nella knowledge base

Preparazione dei dati

1. Individua la `assets/knowledgebase_data_source/` cartella.
2. Posiziona il tuo set di dati in questa cartella.

Modifiche alla configurazione

1. Apri il file `cdk.json`.

2. Vai al `context/configure/paths/knowledgebase_file_name` campo, quindi aggiornalo di conseguenza.
3. Passa al `bedrock_instructions/knowledgebase_instruction` campo, quindi aggiornalo per riflettere accuratamente le sfumature e il contesto del nuovo set di dati.

Per l'integrazione strutturale dei dati

Organizzazione dei dati

1. All'interno della `assets/data_query_data_source/` directory, crea una sottodirectory, ad esempio`tabular_data`.
2. Inserisci il tuo set di dati strutturato (i formati accettabili includono CSV, JSON, ORC e Parquet) in questa sottocartella appena creata.
3. Se ti stai connettendo a un database esistente, aggiorna la funzione per connetterti `create_sql_engine()` `code/lambda/action-lambda/build_query_engine.py` al tuo database.

Aggiornamenti della configurazione e del codice

1. Nel `cdk.json` file, aggiorna il `context/configure/paths/athena_table_data_prefix` campo per allinearlo al nuovo percorso dei dati.
2. Esegui la revisione `code/lambda/action-lambda/dynamic_examples.csv` incorporando nuovi text-to-SQL esempi che corrispondono al tuo set di dati.
3. Esegui la revisione `code/lambda/action-lambda/prompt_templates.py` per rispecchiare gli attributi del tuo set di dati strutturato.
4. Nel `cdk.json` file, aggiorna il `context/configure/bedrock_instructions/action_group_description` campo per spiegare lo scopo e la funzionalità della funzione `Action group` Lambda.
5. Nel `assets/agent_api_schema/artifacts_schema.json` file, spiega le nuove funzionalità della tua funzione `Action group` Lambda.

Aggiornamento generale

Nel `cdk.json` file, nella `context/configure/bedrock_instructions/agent_instruction` sezione, fornisci una descrizione completa della funzionalità e dello scopo di progettazione previsti per l'agente Amazon Bedrock, tenendo conto dei nuovi dati integrati.

Documenta le conoscenze istituzionali utilizzando input vocali utilizzando Amazon Bedrock e Amazon Transcribe

Praveen Kumar Jeyarajan, Jundong Qiao, Rajiv Upadhyay e Megan Wu, Amazon Web Services

Riepilogo

L'acquisizione della conoscenza istituzionale è fondamentale per garantire il successo e la resilienza organizzativi. La conoscenza istituzionale rappresenta la saggezza, le intuizioni e le esperienze collettive accumulate dai dipendenti nel tempo, spesso di natura tacita e tramandate in modo informale. Questa ricchezza di informazioni comprende approcci unici, best practice e soluzioni a problemi complessi che potrebbero non essere documentati altrove. Formalizzando e documentando queste conoscenze, le aziende possono preservare la memoria istituzionale, promuovere l'innovazione, migliorare i processi decisionali e accelerare le curve di apprendimento per i nuovi dipendenti. Inoltre, promuove la collaborazione, dà potere alle persone e coltiva una cultura del miglioramento continuo. In definitiva, lo sfruttamento delle conoscenze istituzionali aiuta le aziende a utilizzare la loro risorsa più preziosa, l'intelligenza collettiva della forza lavoro, per affrontare le sfide, promuovere la crescita e mantenere il vantaggio competitivo in ambienti aziendali dinamici.

Questo modello spiega come acquisire le conoscenze istituzionali attraverso le registrazioni vocali dei dipendenti senior. Utilizza Amazon Transcribe e Amazon Bedrock per la documentazione e la verifica sistematiche. Documentando queste conoscenze informali, puoi preservarle e condividerle con le successive coorti di dipendenti. Questo impegno sostiene l'eccellenza operativa e migliora l'efficacia dei programmi di formazione attraverso l'integrazione delle conoscenze pratiche acquisite attraverso l'esperienza diretta.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Docker, installato
- AWS Cloud Development Kit (AWS CDK) versione 2.114.1 o successiva, installato e avviato nelle regioni AWS `us-east-1` `us-west-2`
- AWS CDK Toolkit versione 2.114.1 o successiva, installato

- AWS Command Line Interface (AWS CLI), installata e configurata
- Python versione 3.12 o successiva, installata
- Autorizzazioni per creare risorse Amazon Transcribe, Amazon Bedrock, Amazon Simple Storage Service (Amazon S3) e AWS Lambda

Limitazioni

- Questa soluzione viene distribuita su un singolo account AWS.
- Questa soluzione può essere distribuita solo nelle regioni AWS in cui sono disponibili Amazon Bedrock e Amazon Transcribe. Per informazioni sulla disponibilità, consulta la documentazione per Amazon Bedrock e Amazon Transcribe.
- I file audio devono essere in un formato supportato da Amazon Transcribe. Per un elenco dei formati supportati, consulta Formati multimediali nella documentazione Transcribe.

Versioni del prodotto

- SDK AWS per Python (Boto3) versione 1.34.57 o successiva
- LangChain versione 0.1.12 o successiva

Architecture

L'architettura rappresenta un flusso di lavoro serverless su AWS. AWS Step Functions orchestra le funzioni Lambda per l'elaborazione audio, l'analisi del testo e la generazione di documenti. Il diagramma seguente mostra il flusso di lavoro Step Functions, noto anche come macchina a stati.

Ogni fase della macchina a stati è gestita da una funzione Lambda distinta. Di seguito sono riportati i passaggi del processo di generazione dei documenti:

1. La funzione `preprocess` Lambda convalida l'input passato a Step Functions ed elenca tutti i file audio presenti nel percorso della cartella URI Amazon S3 fornito. Le funzioni Lambda a valle del flusso di lavoro utilizzano l'elenco di file per convalidare, riepilogare e generare il documento.
2. La funzione `transcribe` Lambda utilizza Amazon Transcribe per convertire i file audio in trascrizioni di testo. Questa funzione Lambda è responsabile dell'avvio del processo di trascrizione e della trasformazione accurata del parlato in testo, che viene quindi memorizzato per l'elaborazione successiva.

3. La funzione `validate` Lambda analizza le trascrizioni del testo, determinando la pertinenza delle risposte alle domande iniziali. Utilizzando un modello linguistico di grandi dimensioni (LLM) tramite Amazon Bedrock, identifica e separa le risposte sull'argomento da quelle fuori tema.
4. La funzione `summarize` Lambda utilizza Amazon Bedrock per generare un riepilogo coerente e conciso delle risposte sull'argomento.
5. La funzione `generate` Lambda riunisce i riepiloghi in un documento ben strutturato. Può formattare il documento in base a modelli predefiniti e includere qualsiasi contenuto o dato aggiuntivo necessario.
6. Se una delle funzioni Lambda non funziona, ricevi una notifica e-mail tramite Amazon Simple Notification Service (Amazon SNS).

Durante questo processo, AWS Step Functions si assicura che ogni funzione Lambda venga avviata nella sequenza corretta. Questa macchina a stati ha la capacità di elaborazione parallela per migliorare l'efficienza. Un bucket Amazon S3 funge da archivio di archiviazione centrale, supportando il flusso di lavoro gestendo i vari formati multimediali e di documenti coinvolti.

Tools (Strumenti)

Servizi AWS

- Amazon Bedrock è un servizio completamente gestito che rende disponibili per l'uso modelli di base ad alte prestazioni (FMs) delle principali startup di intelligenza artificiale e di Amazon tramite un'API unificata.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Step Functions è un servizio di orchestrazione serverless che ti aiuta a combinare le funzioni AWS Lambda e altri servizi AWS per creare applicazioni aziendali critiche.
- Amazon Transcribe è un servizio di riconoscimento vocale automatico che utilizza modelli di apprendimento automatico per convertire l'audio in testo.

Altri strumenti

- LangChainè un framework per lo sviluppo di applicazioni basate su modelli linguistici di grandi dimensioni (LLMs).

Archivio di codice

Il codice per questo pattern è disponibile nel GitHub genai-knowledge-capturerepository.

L'archivio del codice contiene i seguenti file e cartelle:

- `assets`cartella: le risorse statiche per la soluzione, ad esempio il diagramma di architettura e il set di dati pubblico
- `code/lambdas`folder — Il codice Python per tutte le funzioni Lambda
  - `code/lambdas/generate`folder - Il codice Python che genera un documento dai dati riepilogati nel bucket S3
  - `code/lambdas/preprocess`folder - Il codice Python che elabora gli input per la macchina a stati Step Functions
  - `code/lambdas/summarize`folder - Il codice Python che riassume i dati trascritti utilizzando il servizio Amazon Bedrock
  - `code/lambdas/transcribe`folder - Il codice Python che converte i dati vocali (file audio) in testo utilizzando Amazon Transcribe
  - `code/lambdas/validate`folder - Il codice Python che verifica se tutte le risposte riguardano lo stesso argomento
- `code/code_stack.py`— Il file di costruzione in Python del CDK AWS utilizzato per creare risorse AWS
- `app.py`— Il file Python dell'app AWS CDK utilizzato per distribuire le risorse AWS nell'account AWS di destinazione
- `requirements.txt`— L'elenco di tutte le dipendenze Python che devono essere installate per il CDK AWS
- `cdk.json`— Il file di input per fornire i valori necessari per creare risorse

Best practice

L'esempio di codice fornito è destinato esclusivamente a scopi proof-of-concept (PoC) o pilota. Se desideri portare la soluzione in produzione, utilizza le seguenti best practice:

- Abilita la registrazione degli accessi ad Amazon S3
- Abilita i log di flusso VPC

Epiche

Configura le credenziali AWS sulla tua workstation locale

Operazione	Description	Competenze richieste

Esporta le variabili per l'account e la regione AWS.	Per fornire le credenziali AWS per il CDK AWS utilizzando variabili di ambiente, esegui i seguenti comandi.
	`export CDK_DEFAU LT_ACCOUNT=<12-digit AWS account number>`
	`export CDK_DEFAU LT_REGION=<Region>`
		AWS DevOps, DevOps ingegnere

Configura il profilo denominato AWS CLI.	Per configurare il profilo denominato AWS CLI per l'account, segui le istruzioni in Configurazione e impostazioni del file di credenziali.	AWS DevOps, DevOps ingegnere

Configurare l'ambiente

Operazione	Description	Competenze richieste
Clona il repository sulla tua workstation locale.	Per clonare il genai-knowledge-capturerepository,	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	esegui il seguente comando nel tuo terminale. `git clone https://github.com/aws-samples/genai-knowledge-capture`	
(Facoltativo) Sostituisci i file audio.	Per personalizzare l'applicazione di esempio in modo da incorporare i tuoi dati, procedi come segue: 1. Accedere alla `assets/audio_samples` cartella nel repository clonato. 2. Eliminate le cartelle contenenti i file audio di esempio. 3. Crea una cartella per ogni argomento che desideri analizzare. 4. Trasferisci i tuoi file audio nelle rispettive cartelle.	AWS DevOps, DevOps ingegnere

Configura l'ambiente virtuale Python.	Per configurare l'ambiente virtuale Python, esegui i seguenti comandi.
	`cd genai-knowledge-capture`
	`python3 -m venv .venv`
	`source .venv/bin/activate`
	`pip install -r requirements.txt`
		AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
Sintetizza il codice CDK AWS.	Per convertire il codice in una configurazione CloudForm ation dello stack AWS, esegui il comando seguente. `cdk synth`	AWS DevOps, DevOps ingegnere

Configura e distribuisci la soluzione

Operazione	Description	Competenze richieste
Fornire l'accesso al modello di base.	Abilita l'accesso al modello Anthropic Claude 3 Sonnet per il tuo account AWS. Per istruzioni, consulta Aggiunger e l'accesso al modello nella documentazione di Bedrock.	AWS DevOps
Distribuisci risorse nell'acco unt.	Per distribuire risorse nell'acco unt AWS utilizzando il CDK AWS, procedi come segue: 1. (Facoltativo) Nella radice del repository clonato, nel `app.py` file, aggiorna il nome dello stack CloudFormation AWS. Il nome dello stack predefini to è. `genai-knowledge-capture-stack` 2. Per distribuire risorse, esegui il comando. `cdk deploy`	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	Il `cdk deploy` comando utilizza costrutti di livello 3 per creare un set di funzioni Lambda, un bucket S3, un argomento Amazon SNS e una macchina a stati Step Functions. I file audio nella `assets/audio_samples` cartella vengono copiati nel bucket S3 durante la distribuzione. 3. Accedi alla Console di gestione AWS, quindi apri la CloudFormation console all'indirizzo https://console.aws.amazon.com/cloudformation/. 4. Verifica che lo stack sia stato distribuito correttamente. Per istruzioni, consulta Revisione dello stack sulla CloudFormation console AWS.	

Operazione	Description	Competenze richieste
Iscriviti all'argomento Amazon SNS.	Per iscriverti all'argomento di notifica di Amazon SNS, procedi come segue: 1. Nella CloudFormation console, nel riquadro di navigazione, scegli Stacks. 2. Scegli lo `genai-knowledge-capture-stack` stack. 3. Seleziona la scheda Outputs (Output). 4. Trova il nome dell'argomento di Amazon SNS con la chiave. `SNSTopicName` 5. Configura un indirizzo e-mail per ricevere notifiche seguendo le istruzioni riportate nell'argomento Sottoscrivere un indirizzo e-mail a un argomento Amazon SNS.	Informazioni generali su AWS

Test della soluzione

Operazione	Description	Competenze richieste
Avvia la macchina a stati.	1. Apri la console Step Functions. 2. Nella pagina State machines, scegli genai-knowledge-capture-stack-state-machine.	Sviluppatore di app, General AWS

Operazione	Description	Competenze richieste
	3. Selezionare Start execution (Avvia esecuzione). 4. (Facoltativo) Nella casella Nome, inserite un nome per l'esecuzione. 5. Nell'area di input, inserite il seguente oggetto JSON sostituendo il testo segnaposto, dove: • `<Name>` è il nome a cui desiderate assegnare al documento. • `<S3 bucket name>` è il nome del bucket Amazon S3 che contiene i file audio. • `<Folder path>` è la directory che contiene i file audio. `{ "documentName": "<Name>", "audioFileFolderUri": "s3://<S3 bucket name>/<Folder path>" }` 6. Selezionare Start Execution (Avvia esecuzione). 7. Nella pagina dei dettagli dell'esecuzione, esamina i risultati e attendi il completamento dell'esecuzione.	

Pulisci tutte le risorse AWS nella soluzione

Operazione	Description	Competenze richieste
Rimuovi le risorse AWS.	Dopo aver testato la soluzione , pulisci le risorse: 1. Elimina tutti gli oggetti dal bucket S3, quindi elimina il bucket. Per ulteriori informazioni, consulta Eliminazione di un bucket. 2. Dal repository clonato, esegui il comando. `cdk destroy`	AWS DevOps, DevOps ingegnere

Risorse correlate

Documentazione AWS

- Risorse Amazon Bedrock:
  - Accesso al modello
  - Parametri di inferenza per i modelli di base
- Risorse CDK AWS:
  - Inizia a usare il CDK AWS
  - Lavorare con il CDK AWS in Python
  - Risoluzione dei problemi più comuni di AWS CDK
  - Comandi Toolkit
- Risorse AWS Step Functions:
  - Guida introduttiva a AWS Step Functions
  - Risoluzione dei problemi
- Creazione di funzioni Lambda con Python
- Generative AI Application Builder su AWS

Altre risorse

- LangChain documentazione

Genera consigli personalizzati e riclassificati con Amazon Personalize

Mason Cahill, Matthew Chasse e Tayo Olajide, Amazon Web Services

Riepilogo

Questo modello mostra come utilizzare Amazon Personalize per generare consigli personalizzati, inclusi consigli riclassificati, per i tuoi utenti in base all'acquisizione di dati di interazione utente in tempo reale da tali utenti. Lo scenario di esempio utilizzato in questo modello si basa su un sito Web dedicato all'adozione di animali domestici che genera consigli per gli utenti in base alle loro interazioni (ad esempio, quali animali domestici visita un utente). Seguendo lo scenario di esempio, impari a utilizzare Amazon Kinesis Data Streams per importare dati di interazione, AWS Lambda per generare consigli e riclassificarli e Amazon Data Firehose per archiviare i dati in un bucket Amazon Simple Storage Service (Amazon S3). Imparerai anche a usare AWS Step Functions per creare una macchina a stati che gestisca la versione della soluzione (ovvero un modello addestrato) che genera i tuoi consigli.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo con un AWS Cloud Development Kit (AWS CDK) avviato
- AWS Command Line Interface (AWS CLI) con credenziali configurate
- Python 3.9

Versioni del prodotto

- Python 3.9
- AWS CDK 2.23.0 o versione successiva
- AWS CLI 2.7.27 o versione successiva

Architecture

Stack tecnologico

- Amazon Data Firehose

- Flusso di dati Amazon Kinesis
- Amazon Personalize
- Amazon Simple Storage Service (Amazon S3)
- AWS Cloud Development Kit (CDK AWS)
- Interfaccia a riga di comando di AWS (CLI AWS)
- AWS Lambda
- AWS Step Functions

Architettura Target

Il diagramma seguente illustra una pipeline per l'acquisizione di dati in tempo reale in Amazon Personalize. La pipeline utilizza quindi tali dati per generare consigli personalizzati e riclassificati per gli utenti.

Il diagramma mostra il flusso di lavoro seguente:

1. Kinesis Data Streams acquisisce i dati degli utenti in tempo reale (ad esempio, eventi come animali domestici visitati) per l'elaborazione da parte di Lambda e Firehose.
2. Una funzione Lambda elabora i record di Kinesis Data Streams ed effettua una chiamata API per aggiungere l'interazione dell'utente nel record a un tracker di eventi in Amazon Personalize.
3. Una regola basata sul tempo richiama una macchina a stati Step Functions e genera nuove versioni della soluzione per i modelli di raccomandazione e riclassificazione utilizzando gli eventi dell'event tracker in Amazon Personalize.
4. Le campagne Amazon Personalize vengono aggiornate dalla macchina a stati per utilizzare la nuova versione della soluzione.
5. Lambda riordina l'elenco degli articoli consigliati avviando la campagna di riclassificazione di Amazon Personalize.
6. Lambda recupera l'elenco degli articoli consigliati chiamando la campagna di consigli di Amazon Personalize.
7. Firehose salva gli eventi in un bucket S3 dove è possibile accedervi come dati storici.

Tools (Strumenti)

Strumenti AWS

- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software che aiuta a definire e fornire l'infrastruttura cloud AWS in codice.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.
- Amazon Data Firehose ti aiuta a distribuire dati di streaming in tempo reale ad altri servizi AWS, endpoint HTTP personalizzati ed endpoint HTTP di proprietà di provider di servizi terzi supportati.
- Amazon Kinesis Data Streams ti aiuta a raccogliere ed elaborare grandi flussi di record di dati in tempo reale.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- Amazon Personalize è un servizio di machine learning (ML) completamente gestito che ti aiuta a generare consigli sugli articoli per i tuoi utenti in base ai tuoi dati.
- AWS Step Functions è un servizio di orchestrazione senza server che ti aiuta a combinare funzioni Lambda e altri servizi AWS per creare applicazioni aziendali critiche.

Altri strumenti

- pytest è un framework Python per scrivere test piccoli e leggibili.
- Python è un linguaggio di programmazione per computer generico.

Codice

Il codice per questo modello è disponibile nel repository GitHub Animal Recommender. Puoi utilizzare il CloudFormation modello AWS di questo repository per distribuire le risorse per la soluzione di esempio.

> **Note**
>
> Le versioni della soluzione Amazon Personalize, l'event tracker e le campagne sono supportate da risorse personalizzate (all'interno dell'infrastruttura) che si espandono su risorse native. CloudFormation

Epiche

Crea l'infrastruttura

Operazione	Description	Competenze richieste
Crea un ambiente Python isolato.	Configurazione Mac/Linux 1. Per creare manualmente un ambiente virtuale, esegui il `$ python3 -m venv .venv` comando dal tuo terminale. 2. Al termine del processo di inizializzazione, esegui il `$ source .venv/bin/activate` comando per attivare l'ambiente virtuale. Configurazione di Windows Per creare manualmente un ambiente virtuale, esegui il `% .venv\Scripts\activate.bat` comando dal tuo terminale.	DevOps ingegnere
Sintetizza il modello. CloudFormation	1. Per installare le dipendenze richieste, esegui il `$ pip install -r requirements.txt` comando dal tuo terminale. 2. Nella CLI di AWS, imposta le seguenti variabili di ambiente: • `export ACCOUNT_ID=123456789`	DevOps ingegnere

Operazione	Description	Competenze richieste
	• `export CDK_DEPLOY_REGION=us-east-1` • `export CDK_ENVIRONMENT=dev` 3. Nel `config/{env}.yml` file, esegui l'aggiornamento in `vpcId` modo che corrisponda al tuo ID del cloud privato virtuale (VPC). 4. Per sintetizzare il CloudFormation modello per questo codice, esegui il comando. `$ cdk synth` **Note** Nel passaggio 2, `CDK_ENVIRONMENT` fa riferimento al `config/{env}.yml` file.	

Operazione	Description	Competenze richieste
Implementa risorse e crea infrastrutture.	Per distribuire le risorse della soluzione, esegui il `./deploy.sh` comando dal tuo terminale. Questo comando installa le dipendenze Python richieste . Uno script Python crea un bucket S3 e una chiave AWS Key Management Service (AWS KMS), quindi aggiunge i dati iniziali per le creazioni iniziali del modello. Infine, lo script viene eseguito per creare l'infrastruttura rimanente`cdk deploy`. **Note** L'addestramento iniziale del modello avviene durante la creazione dello stack. Il completamento della creazione dello stack può richiedere fino a due ore.	DevOps ingegnere

Risorse correlate

- Consigliere per animali () GitHub
- Documentazione di riferimento per AWS CDK
- Documentazione Boto3

- Ottimizza i consigli personalizzati per una metrica aziendale a tua scelta con Amazon Personalize (AWS Machine Learning Blog)

Informazioni aggiuntive

Esempi di payload e risposte

Raccomandazione: funzione Lambda

Per recuperare i consigli, invia una richiesta alla funzione Lambda delle raccomandazioni con un payload nel seguente formato:

```
{
  "userId": "3578196281679609099",
  "limit": 6
}
```

Il seguente esempio di risposta contiene un elenco di gruppi di animali:

```
[{"id": "1-domestic short hair-1-1"},
{"id": "1-domestic short hair-3-3"},
{"id": "1-domestic short hair-3-2"},
{"id": "1-domestic short hair-1-2"},
{"id": "1-domestic short hair-3-1"},
{"id": "2-beagle-3-3"},
```

Se si omette il `userId` campo, la funzione restituisce raccomandazioni generali.

Riclassificazione della funzione Lambda

Per utilizzare la riclassificazione, invia una richiesta alla funzione di riclassificazione Lambda. Il payload contiene tutti gli elementi da riclassificare e i IDs relativi metadati. `userId` I seguenti dati di esempio utilizzano le classi Oxford Pets for `animal_species_id` (1=cat, 2=dog) e i numeri interi 1-5 per e: `animal_age_id` `animal_size_id`

```
{
   "userId":"12345",
   "itemMetadataList":[
      {
         "itemId":"1",
```

```
         "animalMetadata":{
            "animal_species_id":"2",
            "animal_primary_breed_id":"Saint_Bernard",
            "animal_size_id":"3",
            "animal_age_id":"2"
         }
      },
      {
         "itemId":"2",
         "animalMetadata":{
            "animal_species_id":"1",
            "animal_primary_breed_id":"Egyptian_Mau",
            "animal_size_id":"1",
            "animal_age_id":"1"
         }
      },
      {
         "itemId":"3",
         "animalMetadata":{
            "animal_species_id":"2",
            "animal_primary_breed_id":"Saint_Bernard",
            "animal_size_id":"3",
            "animal_age_id":"2"
         }
      }
   ]
}
```

La funzione Lambda riclassifica questi articoli e quindi restituisce un elenco ordinato che include l'articolo IDs e la risposta diretta di Amazon Personalize. Questo è un elenco classificato dei gruppi di animali a cui appartengono gli articoli e del relativo punteggio. Amazon Personalize utilizza le ricette di personalizzazione degli utenti e di classificazione personalizzata per includere un punteggio per ogni articolo nei consigli. Questi punteggi rappresentano la certezza relativa di Amazon Personalize in merito all'articolo successivo che l'utente sceglierà. I punteggi più alti rappresentano una maggiore certezza.

```
{
   "ranking":[
      "1",
      "3",
      "2"
   ],
```

```
   "personalizeResponse":{
      "ResponseMetadata":{
         "RequestId":"a2ec0417-9dcd-4986-8341-a3b3d26cd694",
         "HTTPStatusCode":200,
         "HTTPHeaders":{
            "date":"Thu, 16 Jun 2022 22:23:33 GMT",
            "content-type":"application/json",
            "content-length":"243",
            "connection":"keep-alive",
            "x-amzn-requestid":"a2ec0417-9dcd-4986-8341-a3b3d26cd694"
         },
         "RetryAttempts":0
      },
      "personalizedRanking":[
         {
            "itemId":"2-Saint_Bernard-3-2",
            "score":0.8947961
         },
         {
            "itemId":"1-Siamese-1-1",
            "score":0.105204
         }
      ],
      "recommendationId":"RID-d97c7a87-bd4e-47b5-a89b-ac1d19386aec"
   }
}
```


Carico utile Amazon Kinesis

Il payload da inviare ad Amazon Kinesis ha il seguente formato:


```
{
    "Partitionkey": "randomstring",
    "Data": {
        "userId": "12345",
        "sessionId": "sessionId4545454",
        "eventType": "DetailView",
        "animalMetadata": {
            "animal_species_id": "1",
            "animal_primary_breed_id": "Russian_Blue",
            "animal_size_id": "1",
            "animal_age_id": "2"
        },
        "animal_id": "98765"
```

```
    }
}
```

> **Note**
>
> Il `userId` campo viene rimosso per un utente non autenticato.

Semplifica i flussi di lavoro di machine learning dallo sviluppo locale agli esperimenti scalabili utilizzando SageMaker AI e Hydra

David Sauerwein, Marco Geiger e Julian Ferdinand Grueber, Amazon Web Services

Riepilogo

Questo modello fornisce un approccio unificato alla configurazione e all'esecuzione di algoritmi di machine learning (ML) dai test locali alla produzione su Amazon AI. SageMaker Gli algoritmi ML sono al centro di questo modello, ma il suo approccio si estende all'ingegneria delle funzionalità, all'inferenza e a intere pipeline ML. Questo modello dimostra la transizione dallo sviluppo di script locali ai lavori di formazione sull' SageMaker intelligenza artificiale attraverso un esempio di caso d'uso.

Un tipico flusso di lavoro ML consiste nello sviluppare e testare soluzioni su un computer locale, eseguire esperimenti su larga scala (ad esempio, con parametri diversi) nel cloud e distribuire la soluzione approvata nel cloud. Quindi, la soluzione implementata deve essere monitorata e gestita. Senza un approccio unificato a questo flusso di lavoro, gli sviluppatori devono spesso rifattorizzare il codice in ogni fase. Se la soluzione dipende da un gran numero di parametri che potrebbero cambiare in qualsiasi fase di questo flusso di lavoro, può diventare sempre più difficile rimanere organizzati e coerenti.

Questo modello affronta queste sfide. Innanzitutto, elimina la necessità di rifattorizzare il codice tra gli ambienti fornendo un flusso di lavoro unificato che rimane coerente indipendentemente dal fatto che venga eseguito su macchine locali, in contenitori o sull'intelligenza artificiale. SageMaker In secondo luogo, semplifica la gestione dei parametri tramite il sistema di configurazione di Hydra, in cui i parametri sono definiti in file di configurazione separati che possono essere facilmente modificati e combinati, con registrazione automatica della configurazione di ogni esecuzione. Per ulteriori dettagli su come questo modello affronta queste sfide, consulta Informazioni aggiuntive.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un ruolo utente AWS Identity and Access Management (IAM) per l'implementazione e l'avvio dei lavori di formazione sull' SageMaker intelligenza artificiale

- AWS Command Line Interface (AWS CLI) versione 2.0 o successiva installata e configurata
- Installata la versione 1.8 o successiva di Poetry, ma precedente alla 2.0
- Docker installato
- Python versione 3.10.x

Limitazioni

- Il codice attualmente si rivolge solo ai lavori di formazione sull' SageMaker intelligenza artificiale. Estenderlo ai processi di elaborazione e a intere pipeline di SageMaker intelligenza artificiale è semplice.
- Per una configurazione SageMaker AI completamente prodotta, è necessario disporre di ulteriori dettagli. Alcuni esempi potrebbero essere le chiavi custom AWS Key Management Service (AWS KMS) per l'elaborazione e lo storage o le configurazioni di rete. È inoltre possibile configurare queste opzioni aggiuntive utilizzando Hydra in una sottocartella dedicata della cartella. `config`
- Alcune Servizi AWS non sono disponibili in tutte. Regioni AWS Per la disponibilità regionale, vedi AWS Servizi per regione. Per endpoint specifici, consulta Service endpoints and quotas e scegli il link relativo al servizio.

Architecture

Il diagramma seguente illustra l'architettura della soluzione.

Il diagramma mostra il flusso di lavoro seguente:

1. Il data scientist può eseguire iterazioni sull'algoritmo su piccola scala in un ambiente locale, regolare i parametri e testare rapidamente lo script di addestramento senza bisogno di Docker o AI. SageMaker (Per maggiori dettagli, consulta l'attività «Esegui localmente per test rapidi» in Epics.)
2. Una volta soddisfatto dell'algoritmo, il data scientist crea e invia l'immagine Docker al repository Amazon Elastic Container Registry (Amazon ECR) denominato. `hydra-sm-artifact` (Per maggiori dettagli, consulta «Esegui flussi di lavoro sull' SageMaker intelligenza artificiale» in Epics.)
3. Il data scientist avvia lavori di formazione sull' SageMaker intelligenza artificiale o lavori di ottimizzazione degli iperparametri (HPO) utilizzando script Python. Per i normali lavori di formazione, la configurazione modificata viene scritta nel bucket Amazon Simple Storage Service

(Amazon S3) denominato. `hydra-sample-config` Per i lavori HPO, viene applicato il set di configurazione predefinito che si trova nella `config` cartella.

4. Il processo di formazione SageMaker AI estrae l'immagine Docker, legge i dati di input dal bucket Amazon S3 e recupera la configurazione dal bucket `hydra-sample-data` Amazon S3 o utilizza la configurazione predefinita. `hydra-sample-config` Dopo l'addestramento, il processo salva i dati di output nel bucket Amazon S3. `hydra-sample-data`

Automazione e scalabilità

- Per la formazione, la riqualificazione o l'inferenza automatizzate, puoi integrare il AWS CLI codice con servizi come AWS LambdaAWS CodePipeline, o Amazon. EventBridge
- La scalabilità può essere ottenuta modificando le configurazioni, ad esempio le dimensioni, o aggiungendo configurazioni per la formazione distribuita.

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita tra e. Account AWS Regioni AWS
- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando. Per questo modello, AWS CLI è utile sia per la configurazione iniziale delle risorse che per il test.
- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- Amazon SageMaker AI è un servizio di machine learning (ML) gestito che ti aiuta a creare e addestrare modelli di machine learning per poi distribuirli in un ambiente ospitato pronto per la produzione. SageMaker AI Training è un servizio di machine learning completamente gestito all'interno dell' SageMaker intelligenza artificiale che consente la formazione di modelli di machine learning su larga scala. Lo strumento è in grado di gestire le esigenze computazionali dei modelli di formazione in modo efficiente, utilizzando la scalabilità e l'integrazione integrate con altri. Servizi AWS SageMaker AI Training supporta anche algoritmi e contenitori personalizzati, rendendolo flessibile per un'ampia gamma di flussi di lavoro ML.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri strumenti

- Docker è un insieme di prodotti Platform as a Service (PaaS) che utilizzano la virtualizzazione a livello di sistema operativo per fornire software in container. È stato utilizzato secondo questo schema per garantire ambienti coerenti in varie fasi, dallo sviluppo alla distribuzione, e per impacchettare dipendenze e codice in modo affidabile. La containerizzazione di Docker ha consentito una facile scalabilità e il controllo delle versioni in tutto il flusso di lavoro.
- Hydra è uno strumento di gestione della configurazione che offre flessibilità per la gestione di più configurazioni e la gestione dinamica delle risorse. È fondamentale per la gestione delle configurazioni degli ambienti e consente un'implementazione senza interruzioni in ambienti diversi. Per ulteriori dettagli su Hydra, vedere Informazioni aggiuntive.
- Python è un linguaggio di programmazione per computer generico. Python è stato usato per scrivere il codice ML e il flusso di lavoro di distribuzione.
- Poetry è uno strumento per la gestione delle dipendenze e la creazione di pacchetti in Python.

Deposito di codice

Il codice per questo pattern è disponibile nel repository GitHub configuring-sagemaker-training-jobs-with-hydra.

Best practice

- Scegli un ruolo IAM per l'implementazione e l'avvio dei lavori di formazione sull' SageMaker intelligenza artificiale che segua il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.
- Utilizza credenziali temporanee per accedere al ruolo IAM nel terminale.

Epiche

Configura l'ambiente

Operazione	Description	Competenze richieste
Crea e attiva l'ambiente virtuale.	Per creare e attivare l'ambient e virtuale, esegui i seguenti	Informazioni generali su AWS

Operazione	Description	Competenze richieste

	comandi nella radice del repository:
	`poetry install`
	`poetry shell`	

Implementa l'infrastruttura.	Per distribuire l'infrastruttura utilizzando CloudFormation, esegui il comando seguente: `aws cloudformation deploy --template-file infra/hydra-sagemaker-setup.yaml --stack-name hydra-sagemaker-setup --capabilities CAPABILITY_NAMED_IAM`	General AWS, DevOps ingegnere
Scarica i dati di esempio.	Per scaricare i dati di input da openml sul computer locale, esegui il seguente comando: `python scripts/download_data.py`	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Esegui localmente per test rapidi.	Per eseguire il codice di addestramento localmente per il test, esegui il comando seguente: `python mypackage/train.py data.train_data_path=data/train.csv evaluation.base_dir_path=data` I registri di tutte le esecuzioni vengono archiviati in base alla data di esecuzione in una cartella denominata. `outputs` Per ulteriori informazioni, consulta la sezione «Output» nel GitHub repository. È inoltre possibile eseguire più corsi di formazione in parallelo, con parametri diversi, utilizzando la `--multirun` funzionalità. Per maggiori dettagli, consulta la documentazione di Hydra.	Data Scientist

Esegui flussi di lavoro sull'intelligenza artificiale SageMaker

Operazione	Description	Competenze richieste
Imposta le variabili di ambiente.	Per eseguire il tuo lavoro sull' SageMaker intelligenza artificiale, imposta le seguenti variabili di ambiente, fornendo	Informazioni generali su AWS

Operazione	Description	Competenze richieste

	il tuo ID Regione AWS e il tuo Account AWS ID:
	
	`export ECR_REPO_NAME=hydra-sm-artifact`
	`export image_tag=latest`
	`export AWS_REGION="<your_aws_region>" # for instance, us-east-1`
	`export ACCOUNT_ID="<your_account_id>"`
	`export BUCKET_NAME_DATA=hydra-sample-data-$ACCOUNT_ID`
	`export BUCKET_NAME_CONFIG=hydra-sample-config-$ACCOUNT_ID`
	`export AWS_DEFAULT_REGION=$AWS_REGION`
	`export ROLE_ARN=arn:aws:iam::${ACCOUNT_ID}:role/hydra-sample-sagemaker`
	`export INPUT_DATA_S3_PATH=s3://$BUCKET_NAME_DATA/hydra-on-sm/input/`
	`export OUTPUT_DATA_S3_PATH=s3://$BUCKET_NAME_DATA/hydra-on-sm/output/`	

Operazione	Description	Competenze richieste

Crea e invia un'immagine Docker.	Per creare l'immagine Docker e inviarla al repository Amazon ECR, esegui il seguente comando:
	`chmod +x scripts/create_and_push_image.sh`
	`scripts/create_and_push_image.sh $ECR_REPO_NAME $image_tag $AWS_REGION $ACCOUNT_ID`
	Questa attività presuppone che tu disponga di credenziali valide nel tuo ambiente. L'immagine Docker viene inviata al repository Amazon ECR specificato nella variabile di ambiente nell'attività precedente e viene utilizzata per attivare il contenitore SageMaker AI in cui verrà eseguito il processo di formazione.
		Ingegnere ML, General AWS

Operazione	Description	Competenze richieste
Copia i dati di input su Amazon S3.	Il processo di formazione sull' SageMaker intelligenza artificiale deve raccogliere i dati di input. Per copiare i dati di input nel bucket Amazon S3 per i dati, esegui il seguente comando: `aws s3 cp data/train.csv "${INPUT_DATA_S3_PATH}train.csv"`	Ingegnere dei dati, General AWS

Operazione	Description	Competenze richieste
Invia lavori di formazione sull' SageMaker intelligenza artificiale.	Per semplificare l'esecuzione degli script, specifica i parametri di configurazione predefiniti nel `default.yaml` file. Oltre a garantire la coerenza tra le esecuzioni, questo approccio offre anche la flessibilità necessaria per sostituire facilmente le impostazioni predefinite, se necessario. Fai riferimento al file di esempio seguente: `python scripts/start_sagemaker_training_job.py sagemaker.role_arn=$ROLE_ARN sagemaker.config_s3_bucket=$BUCKET_NAME_CONFIG sagemaker.input_data_s3_path=$INPUT_DATA_S3_PATH sagemaker.output_data_s3_path=$OUTPUT_DATA_S3_PATH`	General AWS, ingegnere ML, Data scientist

Operazione	Description	Competenze richieste
Esegui l'ottimizzazione degli iperparametri SageMaker AI.	L'esecuzione dell'ottimizzazione degli iperparametri SageMaker AI è simile all'invio di un lavoro di formazione AII. SageMaker Tuttavia, lo script di esecuzione differisce in alcuni aspetti importanti, come si può vedere nel file start_sagemaker_hpo_job.py. Gli iperparametri da ottimizzare devono essere trasmessi attraverso il payload boto3, non attraverso un canale di accesso al processo di formazione. Per avviare il processo di ottimizzazione degli iperparametri (HPO), esegui i seguenti comandi: `python scripts/start_sagemaker_hpo_job.py sagemaker.role_arn=$ROLE_ARN sagemaker.config_s3_bucket=$BUCKET_NAME_CONFIG sagemaker.input_data_s3_path=$INPUT_DATA_S3_PATH sagemaker.output_data_s3_path=$OUTPUT_DATA_S3_PATH`	Data Scientist

Risoluzione dei problemi

Problema	Soluzione
Token scaduto	Esporta nuove credenziali AWS .
Mancanza di autorizzazioni IAM	Assicurati di esportare le credenziali di un ruolo IAM che disponga di tutte le autorizzazioni IAM necessarie per implementare il CloudFormation modello e avviare i lavori di formazione sull' SageMaker IA.

Risorse correlate

- Addestra un modello con Amazon SageMaker AI (AWS documentazione)
- Che cos'è l'ottimizzazione degli iperparametri?

Informazioni aggiuntive

Questo modello affronta le seguenti sfide:

Coerenza dallo sviluppo locale alla distribuzione su larga scala: con questo modello, gli sviluppatori possono utilizzare lo stesso flusso di lavoro, indipendentemente dal fatto che utilizzino script Python locali, eseguano contenitori Docker locali, conducano grandi esperimenti SageMaker sull'intelligenza artificiale o implementino in produzione sull'intelligenza artificiale. SageMaker Questa coerenza è importante per i seguenti motivi:

- Iterazione più rapida: consente una rapida sperimentazione locale senza la necessità di grandi aggiustamenti durante la scalabilità.
- Nessun refactoring: il passaggio a esperimenti più ampi sull' SageMaker intelligenza artificiale è semplice e non richiede alcuna revisione della configurazione esistente.
- Miglioramento continuo: lo sviluppo di nuove funzionalità e il miglioramento continuo dell'algoritmo sono semplici perché il codice rimane lo stesso in tutti gli ambienti.

Gestione della configurazione: questo modello utilizza Hydra, uno strumento di gestione della configurazione, per fornire i seguenti vantaggi:

- I parametri sono definiti nei file di configurazione, separati dal codice.
- Diversi set di parametri possono essere scambiati o combinati facilmente.
- Il tracciamento degli esperimenti è semplificato perché la configurazione di ogni esecuzione viene registrata automaticamente.
- Gli esperimenti sul cloud possono utilizzare la stessa struttura di configurazione delle esecuzioni locali, garantendo la coerenza.

Con Hydra, puoi gestire la configurazione in modo efficace, abilitando le seguenti funzionalità:

- Dividi le configurazioni: suddividi le configurazioni del progetto in parti più piccole e gestibili che possono essere modificate indipendentemente. Questo approccio semplifica la gestione di progetti complessi.
- Modifica facilmente i valori predefiniti: modifica rapidamente le configurazioni di base, semplificando la sperimentazione di nuove idee.
- Allinea gli input della CLI e i file di configurazione: combina facilmente gli input della riga di comando con i file di configurazione. Questo approccio riduce il disordine e la confusione, rendendo il progetto più gestibile nel tempo.

Traduci il linguaggio naturale in query DSL for OpenSearch ed Elasticsearch

Tabby Ward, Nicholas Switzer e Breanne Warner, Amazon Web Services

Riepilogo

Questo modello dimostra come utilizzare modelli linguistici di grandi dimensioni (LLMs) per convertire le query in linguaggio naturale in linguaggio specifico del dominio di query (query DSL), il che semplifica l'interazione degli utenti con servizi di ricerca come Elasticsearch senza una conoscenza approfondita del linguaggio di query. OpenSearch Questa risorsa è particolarmente utile per sviluppatori e data scientist che desiderano potenziare le applicazioni basate sulla ricerca con funzionalità di interrogazione in linguaggio naturale, migliorando in ultima analisi l'esperienza utente e la funzionalità di ricerca.

Il modello illustra le tecniche per una progettazione tempestiva, il perfezionamento iterativo e l'incorporazione di conoscenze specialistiche, tutte fondamentali nella generazione di dati sintetici. Sebbene questo approccio si concentri principalmente sulla conversione delle query, dimostra implicitamente il potenziale dell'aumento dei dati e della produzione scalabile di dati sintetici. Questa base potrebbe essere estesa a attività più complete di generazione di dati sintetici, per evidenziare la potenza del collegamento tra input non strutturati LLMs in linguaggio naturale e output strutturati e specifici dell'applicazione.

Questa soluzione non prevede strumenti di migrazione o implementazione nel senso tradizionale. Si concentra invece sulla dimostrazione di un proof of concept (PoC) per convertire le query in linguaggio naturale in query DSL utilizzando. LLMs

- Il modello utilizza un notebook Jupyter come step-by-step guida per la configurazione dell'ambiente e l'implementazione della conversione. text-to-query
- Utilizza Amazon Bedrock per accedere LLMs, che è fondamentale per interpretare il linguaggio naturale e generare query appropriate.
- La soluzione è progettata per funzionare con Amazon OpenSearch Service. Puoi seguire un processo simile per Elasticsearch e le query generate potrebbero essere potenzialmente adattate a motori di ricerca simili.

Query DSL è un linguaggio di ricerca flessibile basato su JSON che viene utilizzato per costruire query complesse sia in Elasticsearch che. OpenSearch Consente di specificare le query nei

parametri di query delle operazioni di ricerca e supporta vari tipi di query. Una query DSL include query leaf e query composte. Le query Leaf cercano valori specifici in determinati campi e comprendono query full-text, a livello di termine, geografiche, di join, span e specializzate. Le query composte fungono da involucro per più clausole leaf o composte e combinano i risultati o ne modificano il comportamento. Query DSL supporta la creazione di ricerche sofisticate, che vanno da query semplici e corrispondenti a query complesse con più clausole che producono risultati altamente specifici. Query DSL è particolarmente utile per i progetti che richiedono funzionalità di ricerca avanzate, una costruzione flessibile delle query e strutture di query basate su JSON.

Questo modello utilizza tecniche come few-shot prompting, system prompt, output strutturato, concatenamento dei prompt, fornitura di contesto e istruzioni specifiche per le attività per la conversione DSL. text-to-query Per le definizioni e gli esempi di queste tecniche, consultate la sezione Informazioni aggiuntive.

Prerequisiti e limitazioni

Prerequisiti

Per utilizzare efficacemente il notebook Jupyter per convertire le query in linguaggio naturale in query DSL, è necessario:

- Familiarità con i notebook Jupyter. Conoscenze di base su come navigare ed eseguire codice in un ambiente notebook Jupyter.
- Ambiente Python. Un ambiente Python funzionante, preferibilmente Python 3.x, con le librerie necessarie installate.
- Elasticsearch o conoscenza. OpenSearch Conoscenza di base di Elasticsearch o OpenSearch, compresa la sua architettura e come eseguire le query.
- Account AWS. Un utente attivo Account AWS per accedere ad Amazon Bedrock e ad altri servizi correlati.
- Librerie e dipendenze. Installazione di librerie specifiche menzionate nel notebook, ad esempio `boto3` per AWS l'interazione, e qualsiasi altra dipendenza richiesta per l'integrazione LLM.
- Accesso ai modelli all'interno di Amazon Bedrock. Questo modello utilizza tre Claude LLMs di Anthropic. Apri la console Amazon Bedrock e scegli Model access. Nella schermata successiva, scegli Abilita modelli specifici e seleziona questi tre modelli:
  - Claude 3 Sonnet
  - Claude 3.5 Sonnet
  - Claude 3 Haiku

- Politiche IAM adeguate e ruolo IAM. Per eseguire il notebook in un Account AWS ruolo AWS Identity and Access Management (IAM) è necessaria la `SagemakerFullAccess` policy e la policy fornita nella sezione Informazioni aggiuntive, a cui è possibile assegnare un nome`APGtext2querydslpolicy`. Questa politica include l'abbonamento ai tre modelli Claude elencati.

La presenza di questi prerequisiti garantisce un'esperienza fluida quando si lavora con il notebook e si implementano le funzionalità. text-to-query

Limitazioni

- Verifica dello stato del concetto. Questo progetto è inteso principalmente come proof of concept (PoC). Dimostra il potenziale di utilizzo LLMs per convertire le query in linguaggio naturale in query DSL, ma potrebbe non essere completamente ottimizzato o pronto per la produzione.
- Limitazioni del modello:

  Vincoli della finestra contestuale. Quando utilizzi LLMs quelli disponibili su Amazon Bedrock, tieni presente le limitazioni della finestra di contesto:

  Modelli Claude (a settembre 2024):

  - Claude 3 Opus: 200.000 gettoni
  - Claude 3 Sonnet: 200.000 gettoni
  - Claude 3 Haiku: 200.000 gettoni

  Altri modelli su Amazon Bedrock potrebbero avere finestre contestuali di dimensioni diverse. Controlla sempre la documentazione più recente per le informazioni più recenti.

  Disponibilità del modello.La disponibilità di modelli specifici su Amazon Bedrock può variare. Assicurati di avere accesso ai modelli richiesti prima di implementare questa soluzione.
- Limitazioni aggiuntive
  - Complessità delle interrogazioni. L'efficacia del linguaggio naturale per interrogare la conversione DSL può variare a seconda della complessità della query di input.
  - Compatibilità delle versioni. La query DSL generata potrebbe richiedere modifiche in base alla versione specifica di Elasticsearch o OpenSearch in uso.
  - Prestazioni. Questo modello fornisce un'implementazione PoC, quindi la velocità e la precisione di generazione delle query potrebbero non essere ottimali per l'uso in produzione su larga scala.

- Costo. L'utilizzo LLMs in Amazon Bedrock comporta dei costi. Tieni presente la struttura dei prezzi per il modello che hai scelto. Per maggiori informazioni, consulta Prezzi di Amazon Bedrock.
- Manutenzione. Potrebbero essere necessari aggiornamenti regolari dei prompt e della selezione dei modelli per tenere il passo con i progressi della tecnologia LLM e le modifiche alla sintassi DSL delle query.

Versioni del prodotto

Questa soluzione è stata testata in Amazon OpenSearch Service. Se desideri utilizzare Elasticsearch, potresti dover apportare alcune modifiche per replicare l'esatta funzionalità di questo modello.

- OpenSearch compatibilità delle versioni. OpenSearch mantiene la compatibilità con le versioni precedenti all'interno delle versioni principali. Esempio:
  - OpenSearch I client 1.x sono generalmente compatibili con i cluster OpenSearch 1.x.
  - OpenSearch I client 2.x sono generalmente compatibili con i cluster 2.x. OpenSearch

  Tuttavia, quando possibile, è sempre preferibile utilizzare la stessa versione secondaria sia per il client che per il cluster.
- OpenSearch Compatibilità con le API. OpenSearch mantiene la compatibilità delle API con Elasticsearch OSS 7.10.2 per la maggior parte delle operazioni. Tuttavia, esistono alcune differenze, specialmente nelle versioni più recenti.
- OpenSearch considerazioni sull'aggiornamento:
  - I downgrade diretti non sono supportati. Se necessario, utilizzate le istantanee per il rollback.
  - Quando esegui l'aggiornamento, controlla la matrice di compatibilità e le note di rilascio per eventuali modifiche sostanziali.

Considerazioni su Elasticsearch

- Versione Elasticsearch. La versione principale di Elasticsearch che stai utilizzando è fondamentale, perché la sintassi e le funzionalità delle query possono cambiare tra le versioni principali. Attualmente, l'ultima versione stabile è Elasticsearch 8.x. Assicurati che le tue query siano compatibili con la tua versione specifica di Elasticsearch.

- Versione della libreria DSL di Elasticsearch query. Se stai usando la libreria DSL Python di query Elasticsearch, assicurati che la sua versione corrisponda alla tua versione di Elasticsearch. Esempio:
  - Per Elasticsearch 8.x, usa una `elasticsearch-dsl` versione maggiore o uguale alla 8.0.0 ma inferiore alla 9.0.0.
  - Per Elasticsearch 7.x, usa una `elasticsearch-dsl` versione maggiore o uguale a 7.0.0 ma inferiore alla 8.0.0.
- Versione della libreria client. Che tu stia utilizzando il client Elasticsearch ufficiale o un client specifico per una lingua, assicurati che sia compatibile con la tua versione di Elasticsearch.
- Interroga la versione DSL. Query DSL si evolve con le versioni di Elasticsearch. Alcuni tipi o parametri di query potrebbero essere obsoleti o introdotti in versioni diverse.
- Versione di mappatura. Il modo in cui si definiscono le mappature per gli indici e si passa da una versione all'altra. Controlla sempre la documentazione di mappatura per la tua versione specifica di Elasticsearch.
- Versioni degli strumenti di analisi. Se utilizzi analizzatori, tokenizzatori o altri strumenti di analisi del testo, il loro comportamento o la loro disponibilità potrebbero cambiare tra le versioni.

Architecture

Architettura Target

Il diagramma seguente illustra l'architettura di questo modello.

dove:

1. Input dell'utente e prompt di sistema con alcuni esempi di prompt. Il processo inizia con un utente che fornisce una query in linguaggio naturale o una richiesta per la generazione di schemi.
2. Amazon Bedrock L'input viene inviato ad Amazon Bedrock, che funge da interfaccia per accedere a Claude LLM.
3. Claude 3 Sonnet LLM. Amazon Bedrock utilizza Claude 3 Sonnet della famiglia Claude 3 LLMs per elaborare l'input. Interpreta e genera l'Elasticsearch o la query DSL appropriata. OpenSearch Per le richieste di schema, genera Elasticsearch o mappature sintetiche. OpenSearch
4. Generazione di query DSL. Per le query in linguaggio naturale, l'applicazione prende l'output del LLM e lo formatta in un DSL di query Elasticsearch o Service valido. OpenSearch

5. Generazione di dati sintetici. L'applicazione utilizza anche schemi per creare Elasticsearch sintetici o OpenSearch dati da caricare in una raccolta OpenSearch Serverless per i test.
6. OpenSearch o Elasticsearch. La Query DSL generata viene interrogata su una raccolta OpenSearch Serverless su tutti gli indici. L'output JSON contiene i dati pertinenti e il numero di hit dei dati che risiedono nella raccolta Serverless. OpenSearch

Automazione e scalabilità

Il codice fornito con questo pattern è creato esclusivamente per scopi PoC. L'elenco seguente fornisce alcuni suggerimenti per automatizzare e scalare ulteriormente la soluzione e portare il codice alla produzione. Questi miglioramenti non rientrano nell'ambito di questo modello.

- Containerizzazione:
  - Dockerizza l'applicazione per garantire la coerenza tra diversi ambienti.
  - Utilizza piattaforme di orchestrazione dei container come Amazon Elastic Container Service (Amazon ECS) o Kubernetes per distribuzioni scalabili.
- Architettura serverless:
  - Converti le funzionalità di base in AWS Lambda funzioni.
  - Usa Amazon API Gateway per creare RESTful endpoint per l'input di query in linguaggio naturale.
- Elaborazione asincrona:
  - Implementa Amazon Simple Queue Service (Amazon SQS) per mettere in coda le query in entrata.
  - Utilizzalo AWS Step Functions per orchestrare il flusso di lavoro di elaborazione delle query e generazione di query DSL.
- Memorizzazione nella cache:
  - Implementa un meccanismo per memorizzare nella cache i prompt.
- Monitoraggio e registrazione:
  - Usa Amazon CloudWatch per il monitoraggio e gli avvisi.
  - Implementa la registrazione centralizzata con Amazon CloudWatch Logs o Amazon OpenSearch Service per l'analisi dei log.
- Miglioramenti della sicurezza:
  - Implementa i ruoli IAM per un controllo granulare degli accessi.

  - Utilizzalo Gestione dei segreti AWS per archiviare e gestire in modo sicuro chiavi e credenziali API.
- Implementazione in più regioni:
  - Prendi in considerazione l'implementazione della soluzione su più sistemi Regioni AWS per migliorare la latenza e il disaster recovery.
  - Usa Amazon Route 53 per il routing intelligente delle richieste.

Implementando questi suggerimenti, puoi trasformare questo PoC in una soluzione robusta, scalabile e pronta per la produzione. Si consiglia di testare a fondo ogni componente e l'intero sistema prima dell'implementazione completa.

Tools (Strumenti)

Strumenti

- I notebook Amazon SageMaker AI sono notebook Jupyter completamente gestiti per lo sviluppo dell'apprendimento automatico. Questo modello utilizza i notebook come ambiente interattivo per l'esplorazione dei dati, lo sviluppo di modelli e la sperimentazione in Amazon AI. SageMaker I notebook offrono una perfetta integrazione con altre funzionalità di intelligenza artificiale e. SageMaker Servizi AWS
- Python è un linguaggio di programmazione per computer generico. Questo modello utilizza Python come linguaggio principale per implementare la soluzione.
- Amazon Bedrock è un servizio completamente gestito che rende disponibili per l'uso modelli di base ad alte prestazioni (FMs) delle principali startup di intelligenza artificiale e di Amazon tramite un'API unificata. Amazon Bedrock fornisce l'accesso all' LLMs elaborazione del linguaggio naturale. Questo modello utilizza modelli Anthropic Claude 3.
- AWS SDK per Python (Boto3)è un kit di sviluppo software che ti aiuta a integrare la tua applicazione, libreria o script Python con Servizi AWS, incluso Amazon Bedrock.
- Amazon OpenSearch Service è un servizio gestito che ti aiuta a distribuire, gestire e scalare i cluster OpenSearch di servizi nel cloud AWS. Questo modello utilizza OpenSearch Service come sistema di destinazione per la generazione di query DSL.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub Prompt Engineering Text-to-QueryDSL Using Claude 3 Models. L'esempio utilizza un'app di social media sulla salute che crea post per utenti e profili utente associati all'applicazione sanitaria.

Best practice

Quando utilizzi questa soluzione, considera quanto segue:

- La necessità di AWS credenziali e autorizzazioni adeguate per accedere ad Amazon Bedrock
- Potenziali costi associati all'utilizzo e Servizi AWS LLMs
- L'importanza di comprendere le query DSL per convalidare e potenzialmente modificare le query generate

Epiche

Configura l'ambiente e prepara i dati

Operazione	Description	Competenze richieste
Configura l'ambiente di sviluppo.	**Note** Per istruzioni dettagliate e codice per questo e gli altri passaggi di questo schema, consulta la procedura dettagliata completa nel GitHub repository. 1. Installa i pacchetti Python richiesti, inclusi**`boto3`**,, **`numpy`** **`awscliopensearch-py`** , e **`requests-aws4auth`** usando pip. 2. Importa i moduli necessari come	Python, pip, SDK AWS

Operazione	Description	Competenze richieste
	`boto3json`,`os`,`opensearchpy` , `opensearch` `RequestsHttpConnection` from`Opensearchpy` , `bulk` from`opensearchpy.helpers` ,`sagemaker` , `time` `randomre`, e `AWS4Auth` from. `requests_aws4auth`	
Configura l'accesso. AWS	Configura il client Amazon Bedrock e la sessione SageMaker AI. Recupera l'Amazon Resource Name (ARN) per SageMaker il ruolo di esecuzione AI per utilizzarlo successivamente nella creazione OpenSearch della raccolta Serverless.	IAM, AWS CLI, Amazon Bedrock, Amazon SageMaker
Carica gli schemi delle app sanitarie.	Leggi e analizza gli schemi JSON per i post sanitari e i profili utente da file predefiniti. Converti gli schemi in stringhe per utilizzarli successivamente nei prompt.	DevOps ingegnere, AWS generale, Python, JSON

Genera dati sintetici

Operazione	Description	Competenze richieste
Crea un generatore di dati basato su LLM.	Implementa la funzione generate_data () per chiamare	Python, API Amazon Bedrock, prompt LLM

Operazione	Description	Competenze richieste

	l'API Amazon Bedrock Converse con i modelli Claude 3. Configura il modello IDs per Sonnet, Sonnet 3.5 e Haiku:
	`model_id_sonnet3_5 = "anthropic.claude-3-5-sonnet-20240620-v1:0"`
	`model_id_sonnet = "anthropic.claude-3-sonnet-20240229-v1:0"`
	`model_id_haiku = "anthropic.claude-3-haiku-20240307-v1:0"`	

Crea post sanitari sintetici.	Utilizzate la funzione generate_data () con un messaggio specifico per creare post sintetici sulla salute in base allo schema fornito. La chiamata alla funzione ha il seguente aspetto: `health_post_data = generate_data(bedrock_rt, model_id_sonnet, system_prompt, message_healthpost, inference_config)`	Python, JSON

Operazione	Description	Competenze richieste
Crea profili utente sintetici.	Utilizzate la funzione generate_data () con un messaggio specifico per creare voci sintetiche del profilo utente in base allo schema fornito. È simile alla generazione di post sanitari, ma utilizza un prompt diverso.	Python, JSON

Configura OpenSearch e acquisisci dati

Operazione	Description	Competenze richieste
Configura una raccolta OpenSearch Serverless.	Usa Boto3 per creare una raccolta OpenSearch Serverless con politiche di crittografia, rete e accesso appropriate. La creazione della collezione ha il seguente aspetto: `collection = aoss_client.create_collection(name=es_name, type='SEARCH')` Per ulteriori informazioni su OpenSearch Serverless, consulta la AWS documentazione.	OpenSearch Serverless, IAM
Definisci gli OpenSearch indici.	Crea indici per postazioni sanitarie e profili utente utilizzando il OpenSearch client, in base alle mappature	OpenSearch, JSON

Operazione	Description	Competenze richieste
	dello schema predefinite. La creazione dell'indice ha il seguente aspetto: `response_health = oss_client.indices.create(healthpost_index, body=healthpost_body)`	
Carica i dati in. OpenSearch	Esegui la funzione ingest_data () per inserire in blocco i post sanitari sintetici e i profili utente nei rispettiv i indici. OpenSearch La funzione utilizza l'helper bulk di: `opensearch-py` `success, failed = bulk(oss_client, actions)`	Python, OpenSearch API, operazioni di dati di massa

Genera ed esegui interrogazioni

Operazione	Description	Competenze richieste
Progetta alcuni esempi di prompt.	Genera query di esempio e domande corrispondenti in linguaggio naturale utilizzando i modelli Claude 3 come esempi di pochi esempi per la generazione di query. Il prompt di sistema include questi esempi:	Richiesta LLM, interrogazione DSL

Operazione	Description	Competenze richieste
	`system_prompt_query_generation = [{"text": f"""You are an expert query dsl generator. ... Examples: {example_prompt} ..."""}]`	
Crea un convertitore DSL text-to-query.	Implementa il prompt di sistema, che include schemi, dati ed esempi sintetici, per la generazione di query. Utilizzate il prompt di sistema per convertire le query in linguaggio naturale in query DSL. La chiamata alla funzione ha il seguente aspetto: `query_response = generate_data(bedrock_rt, model_id, system_prompt_query_generation, query, inference_config)`	Python, API Amazon Bedrock, prompt LLM

Operazione	Description	Competenze richieste
Interrogazione di prova DSL attiva. OpenSearch	Eseguite la funzione query_oss () per eseguire la query DSL generata sulla raccolta OpenSearch Serverless e restituire i risultati . La funzione utilizza il metodo di ricerca del client OpenSearch : `response = oss_client.search(index="_all", body=temp)`	Python, OpenSearch API, interrogazione DSL

Testa e valuta

Operazione	Description	Competenze richieste
Crea un set di query di test.	Usa Claude 3 per generare una serie diversificata di domande di test basate su dati e schemi sintetici: `test_queries = generate_data(bedrock_rt, model_id_sonnet, query_system_prompt, query_prompt, inference_config)`	Richiesta LLM
Valuta l'accuratezza della conversione DSL della query.	Testa la query DSL generata eseguendo query OpenSearch e analizzando i risultati restituiti per verificarne la pertinenza e la precisione. Ciò comporta	Python, analisi dei dati, query DSL

Operazione	Description	Competenze richieste
	l'esecuzione della query e l'ispezione dei risultati: `output = query_oss (response1) print("Response after running query against Opensearch") print(output)`	
Modelli Benchmark Claude 3.	Confronta le prestazioni di diversi modelli Claude 3 (Haiku, Sonnet, Sonnet 3.5) per la generazione di query in termini di precisione e latenza. Per confrontare, modifica il `model_id` momento in cui chiami generate_data () e misura il tempo di esecuzione.	Python, analisi comparativa delle prestazioni

Pulisci e documenta

Operazione	Description	Competenze richieste
Sviluppa un processo di pulizia.	Elimina tutti gli indici dalla raccolta OpenSearch Serverless dopo l'uso.	Python, SDK AWS, API OpenSearch

Risorse correlate

- Interrogazione DSL (documentazione) OpenSearch
- Documentazione OpenSearch del servizio Amazon
- OpenSearch Raccolte serverless
- Documentazione Amazon Bedrock

- Documentazione Amazon SageMaker AI
- AWS SDK per Python (Boto3) documentazione

Informazioni aggiuntive

Policy IAM

Ecco la `APGtext2querydslpolicy` politica per il ruolo IAM utilizzata in questo modello:

```
{
  "Version": "2012-10-17",
  "Statement": [
    { "Effect": "Allow",
      "Action": [
        "bedrock:InvokeModel",
        "bedrock:InvokeModelWithResponseStream"
      ],
      "Resource": "*"
    },
    { "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:PutObject",
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::sagemaker-*",
        "arn:aws:s3:::sagemaker-*/*"
      ]
    },
    { "Effect": "Allow",
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents"
      ],
      "Resource": "arn:aws:logs:*:*:log-group:/aws/sagemaker/*"
    },
    { "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface",
        "ec2:DescribeNetworkInterfaces",
```

```
        "ec2:DeleteNetworkInterface"
      ],
      "Resource": "*"
    },
    { "Effect": "Allow",
      "Action": [
        "aoss:*"
      ],
      "Resource": "*"
    },
    { "Effect": "Allow",
      "Action": [
        "iam:PassRole",
        "sagemaker:*"
      ],
      "Resource": [
        "arn:aws:iam::*:role/*", "*"
      ],
      "Condition": {
        "StringEquals": {
          "iam:PassedToService": "sagemaker.amazonaws.com"
          }
      }
    },
    { "Effect": "Allow",
      "Action": [
        "codecommit:GetBranch",
        "codecommit:GetCommit",
        "codecommit:GetRepository",
        "codecommit:ListBranches",
        "codecommit:ListRepositories"
      ],
      "Resource": "*"
    },
    { "Effect": "Allow",
      "Action": [
        "aws-marketplace:Subscribe"
      ],
      "Resource": "*",
      "Condition": {
        "ForAnyValue:StringEquals": {
          "aws-marketplace:ProductId": [
            "prod-6dw3qvchef7zy",
            "prod-m5ilt4siql27k",
```

```
            "prod-ozonys2hmmpeu"
          ]
        }
      }
    },
    { "Effect": "Allow",
      "Action": [
        "aws-marketplace:Unsubscribe",
        "aws-marketplace:ViewSubscriptions"
      ],
      "Resource": "*"
    },
    { "Effect": "Allow",
      "Action": "iam:*",
      "Resource": "*"
    }
  ]
}
```

Tecniche rapide con i modelli Anthropic Claude 3

Questo modello illustra le seguenti tecniche di richiesta per la conversione text-to-query DSL utilizzando i modelli Claude 3.

- Few-shot prompting: il few-shot prompting è una tecnica potente per migliorare le prestazioni dei modelli Claude 3 su Amazon Bedrock. Questo approccio prevede di fornire al modello un numero limitato di esempi che dimostrino il input/output comportamento desiderato prima di chiedergli di eseguire un'attività simile. Quando utilizzi i modelli Claude 3 su Amazon Bedrock, i suggerimenti in pochi passaggi possono essere particolarmente efficaci per attività che richiedono una formattazione specifica, schemi di ragionamento o conoscenze del dominio. Per implementare questa tecnica, in genere si struttura il prompt con due componenti principali: la sezione di esempio e la query effettiva. La sezione di esempio contiene una o più input/output coppie che illustrano l'attività e la sezione query presenta il nuovo input per il quale si desidera una risposta. Questo metodo aiuta Claude 3 a comprendere il contesto e il formato di output previsto e spesso si traduce in una risposta più accurata e coerente.

  Esempio:

```
"query": {
  "bool": {
    "must": [
```

```
      {"match": {"post_type": "recipe"}},
      {"range": {"likes_count": {"gte": 100}}},
      {"exists": {"field": "media_urls"}}
    ]
  }
}
Question: Find all recipe posts that have at least 100 likes and include media URLs.
```

- Istruzioni di sistema: oltre ai prompt di pochi passaggi, i modelli Claude 3 su Amazon Bedrock supportano anche l'uso dei prompt di sistema. I prompt di sistema sono un modo per fornire contesto generale, istruzioni o linee guida al modello prima di presentarlo con input specifici dell'utente. Sono particolarmente utili per impostare il tono, definire il ruolo del modello o stabilire vincoli per l'intera conversazione. Per utilizzare un prompt di sistema con Claude 3 su Amazon Bedrock, devi includerlo nel `system` parametro della tua richiesta API. Questo è separato dai messaggi degli utenti e si applica all'intera interazione. I prompt di sistema dettagliati vengono utilizzati per impostare il contesto e fornire linee guida per il modello.

  Esempio:

```
You are an expert query dsl generator. Your task is to take an input question and
 generate a query dsl to answer the question. Use the schemas and data below to
 generate the query.

Schemas: [schema details]
Data: [sample data]
Guidelines:
- Ensure the generated query adheres to DSL query syntax
- Do not create new mappings or other items that aren't included in the provided
 schemas.
```

- Output strutturato: è possibile indicare al modello di fornire output in formati specifici, come JSON o all'interno di tag XML.

  Esempio:

```
Put the query in json tags
```

- Concatenamento dei prompt: il notebook utilizza l'output di una chiamata LLM come input per un'altra, ad esempio utilizza dati sintetici generati per creare domande di esempio.

- Fornitura del contesto: il contesto pertinente, inclusi schemi e dati di esempio, viene fornito nelle istruzioni.

  Esempio:

  ```
  Schemas: [schema details]
  Data: [sample data]
  ```

- Istruzioni specifiche per attività: vengono creati diversi prompt per attività specifiche, come la generazione di dati sintetici, la creazione di domande di esempio e la conversione di query in linguaggio naturale in query DSL.

  Esempio di generazione di domande di prova:

  ```
  Your task is to generate 5 example questions users can ask the health app based on
   provided schemas and data. Only include the questions generated in the response.
  ```

Usa Amazon Q Developer come assistente di programmazione per aumentare la tua produttività

Ram Kandaswamy, Amazon Web Services

Riepilogo

Questo modello utilizza un tic-tac-toe gioco per dimostrare come applicare Amazon Q Developer a una serie di attività di sviluppo. Genera codice per un tic-tac-toe gioco come applicazione a pagina singola (SPA), migliora l'interfaccia utente e crea script su cui distribuire l'applicazione. AWS

Amazon Q Developer funge da assistente di codifica per accelerare i flussi di lavoro di sviluppo software e migliorare la produttività sia per gli sviluppatori che per i non sviluppatori. Indipendentemente dalla tua esperienza tecnica, ti aiuta a creare architetture e progettare soluzioni per problemi aziendali, avvia il tuo ambiente di lavoro, ti aiuta a implementare nuove funzionalità e genera casi di test per la convalida. Utilizza istruzioni in linguaggio naturale e funzionalità di intelligenza artificiale per garantire codice coerente e di alta qualità e mitigare le sfide di codifica indipendentemente dalle tue capacità di programmazione.

Il vantaggio principale di Amazon Q Developer è la sua capacità di liberarti da attività di codifica ripetitive. Quando usi l'`@workspace`annotazione, Amazon Q Developer inserisce e indicizza tutti i file di codice, le configurazioni e la struttura del progetto nel tuo ambiente di sviluppo integrato (IDE) e fornisce risposte personalizzate per aiutarti a concentrarti sulla risoluzione creativa dei problemi. Puoi dedicare più tempo alla progettazione di soluzioni innovative e al miglioramento dell'esperienza utente. Se non sei tecnico, puoi utilizzare Amazon Q Developer per semplificare i flussi di lavoro e collaborare in modo più efficace con il team di sviluppo. La funzionalità del codice Amazon Q Developer Explain offre istruzioni e riepiloghi dettagliati, in modo da poter navigare tra basi di codice complesse.

Inoltre, Amazon Q Developer offre un approccio indipendente dal linguaggio che aiuta gli sviluppatori di livello medio e basso ad ampliare le proprie competenze. Puoi concentrarti sui concetti fondamentali e sulla logica aziendale anziché sulla sintassi specifica del linguaggio. Ciò riduce la curva di apprendimento quando si cambia tecnologia.

Prerequisiti e limitazioni

Prerequisiti

- IDE (ad esempio, WebStorm o Visual Studio Code) con il plug-in Amazon Q Developer installato. Per istruzioni, consulta Installazione dell'estensione o del plug-in Amazon Q Developer nel tuo IDE nella documentazione di Amazon Q Developer.
- Una Account AWS configurazione attiva con Amazon Q Developer. Per istruzioni, consulta la sezione Guida introduttiva nella documentazione di Amazon Q Developer.
- npm installato. Per istruzioni, consulta la documentazione di npm. Questo modello è stato testato con la versione 10.8 di npm.
- AWS Command Line Interface (AWS CLI) installato. Per istruzioni, consultate la AWS CLI documentazione.

Limitazioni

- Amazon Q Developer può eseguire solo un'attività di sviluppo alla volta.
- Alcune Servizi AWS non sono disponibili in tutte Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Tools (Strumenti)

- Questo modello richiede un IDE come Visual Studio Code o. WebStorm Per un elenco di quelli supportati IDEs, consulta la documentazione di Amazon Q Developer.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire Servizi AWS tramite comandi nella shell della riga di comando.

Best practice

Consulta le migliori pratiche di codifica con Amazon Q Developer in AWS Prescriptive Guidance. Inoltre:

- Quando fornisci istruzioni ad Amazon Q Developer, assicurati che le istruzioni siano chiare e inequivocabili. Aggiungi frammenti di codice e annotazioni, ad esempio, al prompt `@workspace` per fornire più contesto ai tuoi prompt.
- Includi le librerie pertinenti e importale per evitare conflitti o ipotesi errate da parte del sistema.
- Se il codice generato non è preciso o come previsto, utilizzate l'opzione Fornisci feedback e rigenera. Prova a suddividere le istruzioni in istruzioni più piccole.

Epiche

Configura l'ambiente di lavoro

Operazione	Description	Competenze richieste
Crea un nuovo progetto.	Per creare un nuovo progetto nel tuo ambiente di lavoro, esegui il seguente comando e accetta le impostazioni predefinite per tutte le domande: `npx create-next-app@latest`	Sviluppatore di app, programmatore, sviluppatore di software
Prova l'applicazione di base.	Eseguite il comando seguente e confermate che l'applicazione di base venga caricata correttamente nel browser: `npm run dev`	Sviluppatore di app, programmatore, sviluppatore di software
Pulisci il codice base.	Vai al `page.tsx` file nella `src/app` cartella ed elimina il contenuto predefinito per ottenere una pagina vuota. Dopo l'eliminazione, il file dovrebbe avere il seguente aspetto: `export default function Home() { return (<div></div>); }`	Sviluppatore di app, programmatore, sviluppatore di software

Usa Amazon Q Developer per progettare un progetto di tic-tac-toe gioco

Operazione	Description	Competenze richieste
Ottieni una panoramica dei passaggi.	1. Nel tuo IDE, apri il progetto e scegli l'icona Amazon Q per aprire il pannello di chat. 2. Nel pannello di chat di Amazon Q Developer, richiedi una panoramica sulla creazione di un'applicazione a pagina singola (SPA). Esempio: `I would like to create a single-page application involving Next.js React framework for tic-tac-toe game. What are the steps?`	Sviluppatore di app, programmatore, sviluppatore di software
Genera codice per tic-tac-toe.	Nel pannello di chat, avvia un'attività di sviluppo utilizzando il `/dev` comando seguito dalla descrizione dell'attività. Esempio: `/dev Create a React-based single-page application written in TypeScript for a tic-tac-toe game with the following specifications: 1. Design an aesthetically pleasing interface with the game grid`	Sviluppatore di app, programmatore, sviluppatore di software

Operazione	Description	Competenze richieste
	centered vertically and horizontally on the page. 2. Include a heading and clear instructions on how to play the game. 3. Implement color-coding for X and O marks to distinguish them easily. Amazon Q Developer genera codice in base alle tue istruzioni.	
Ispeziona e accetta il codice generato.	Ispeziona visivamente il codice e scegli Accetta codice per sostituire automaticamente il `page.tsx` file. Se riscontri problemi, scegli Fornisci feedback e rigenera e descrivi il problema che hai riscontrato.	Sviluppatore di app, programmatore, sviluppatore di software

Operazione	Description	Competenze richieste
Correggi gli errori di lint.	Il tic-tac-toe gioco di esempio include una griglia. Il codice generato da Amazon Q Developer potrebbe utilizzar e il tipo predefinitoany. Puoi aggiungere la sicurezza dei tipi richiedendo ad Amazon Q Developer come segue: `/dev Ensure proper TypeScript typing for the onSquare Click event handler to resolve any 'any' type issues.`	Sviluppatore di app, programmatore, sviluppatore di software

Operazione	Description	Competenze richieste

Aggiungi fascino visivo.	È possibile suddividere il requisito originale in frammenti più piccoli. Ad esempio, puoi migliorare l'interfaccia utente del gioco con le seguenti istruzioni nelle attività di sviluppo. Questo prompt migliora gli stili CSS (Cascading Style Sheets) ed esporta l'app per la distribuzione.
	`/dev Debug and fix any CSS issues to correctly display the game grid and overall layout.`
	
	`Simplify the code by removing game history functionality and related components.`
	
	`Implement static file export to an 'out' directory for easy deployment. The solution should be fully functional, visually appealing, and free of typing errors or layout issues.`
		Sviluppatore di app, programmatore, sviluppatore di software

Operazione	Description	Competenze richieste

Prova di nuovo.	1. Ora che hai completato il ciclo di vita dello sviluppo, prova il codice per confermare che funzioni come previsto. Per eseguire l'applicazione localmente, utilizzate il comando:
	`npm run dev`
	2. Se l'applicazione funziona come previsto, utilizzate il `build` comando per esportare l'intera applicazione nella cartella di output in preparazione della distribuzione:
	`npm run build`
		Sviluppatore di app, programmatore, sviluppatore di software

Distribuisci l'applicazione su Cloud AWS

Operazione	Description	Competenze richieste

Crea cartelle e file per la distribuzione.	Nel progetto nel tuo ambiente di lavoro, crea una cartella di distribuzione e due file al suo interno: `pushtos3.sh` `ecloudformation.yml` :
	`mkdir deployment && cd deployment`
	`touch pushtos3.sh && chmod +x pushtos3.sh`
		Sviluppatore di app, programmatore, sviluppatore di software

Operazione	Description	Competenze richieste
	`touch cloudformation.yml`	
Genera codice di automazione.	1. Nel pannello di chat per Amazon Q Developer , fornisci la seguente richiesta: `/dev Generate a Cloudformation template that creates two resources: tictactoe artifact bucket and CloudFront. CloudFront should be configured with this bucket as origin. Add cache policy appropriate for Amazon S3 and default root object as index.html. Ensure that origin access control is used and no public bucket is created. Output all the resources and their ARNs.` 2. Rivedi e accetta il codice generato. Il `cloudformation.yml` file creato in precedenza dovrebbe ora essere popolato con uno CloudFormation script che crea le risorse per. Cloud AWS	Amministratore AWS, AWS DevOps, sviluppatore di app

Operazione	Description	Competenze richieste
Genera contenuti di script.	Per creare uno script di distribuzione, utilizzate il seguente prompt: `/dev Modify the pushtos3 shell script so that it can use AWS CLI commands to create a CloudFormation stack named tictactoe-stack if it does not exist already, and use cloudformation.yml as the source template. Wait for the stack to complete and sync the contents from the out folder to the S3 bucket. Perform invalidation of the CloudFront origin.`	Sviluppatore di app, programmatore, sviluppatore di software
Distribuisci l'applicazione su. Cloud AWS	1. Configura l'ambiente di lavoro con AWS credenziali valide. 2. Esegui lo script della shell per distribuire il tic-tac-t oe gioco completamente funzionante su. Cloud AWS	Amministratore AWS, AWS DevOps, architetto del cloud, sviluppatore di app

risoluzione dei problemi

Problema	Soluzione

La build non crea un'applicazione a pagina singola né la esporta nella cartella di output.	Guarda il contenuto del `next.config.mjs` file.
	
	Se il codice ha la seguente configurazione predefinita:
	
	`const nextConfig = {};`
	
	modificalo come segue:
	
	`const nextConfig = {`
	`  output: 'export',`
	`  distDir: 'out',`
	`};`

Risorse correlate

- Creare un nuovo progetto React (documentazione React)
- Panoramica di Amazon Q per sviluppatori (AWS documentazione)
- Best practice per gli sviluppatori di Amazon Q (Guida AWS prescrittiva)
- Installazione, configurazione e utilizzo di Amazon Q Developer con JetBrains IDEs (YouTube video)
- Installazione di Amazon Q per la riga di comando (AWS documentazione)

Usa SageMaker Processing per l'ingegneria di funzionalità distribuite di set di dati ML su scala terabyte

Chris Boomhower, Amazon Web Services

Riepilogo

Molti set di dati su scala terabyte o più grandi spesso sono costituiti da una struttura gerarchica di cartelle e i file del set di dati a volte condividono interdipendenze. Per questo motivo, gli ingegneri del machine learning (ML) e i data scientist devono prendere decisioni progettuali ponderate per preparare tali dati per l'addestramento e l'inferenza dei modelli. Questo modello dimostra come è possibile utilizzare tecniche manuali di macrosharding e microsharding in combinazione con Amazon SageMaker Processing e la parallelizzazione della CPU virtuale (vCPU) per scalare in modo efficiente i processi di progettazione delle funzionalità per complessi set di dati ML di big data.

Questo modello definisce il macrosharding come la suddivisione di directory di dati su più macchine per l'elaborazione e il microsharding come la suddivisione dei dati su ogni macchina su più thread di elaborazione. Il modello dimostra queste tecniche utilizzando Amazon SageMaker con esempi di record di forme d'onda di serie temporali dal set di dati MIMIC-III. PhysioNet Implementando le tecniche di questo modello, è possibile ridurre al minimo i tempi e i costi di elaborazione per la progettazione delle funzionalità, massimizzando al contempo l'utilizzo delle risorse e l'efficienza della produttività. Queste ottimizzazioni si basano sull' SageMaker elaborazione distribuita su istanze Amazon Elastic Compute Cloud EC2 (Amazon) e v CPUs per set di dati simili e di grandi dimensioni, indipendentemente dal tipo di dati.

Prerequisiti e limitazioni

Prerequisiti

- Accedi alle istanze di SageMaker notebook o a SageMaker Studio, se desideri implementare questo modello per il tuo set di dati. Se utilizzi Amazon SageMaker per la prima volta, consulta la sezione Guida introduttiva ad Amazon SageMaker nella documentazione AWS.
- SageMaker Studio, se desideri implementare questo modello con i dati di esempio PhysioNet MIMIC-III.
- Il pattern utilizza SageMaker Processing, ma non richiede alcuna esperienza nell'esecuzione SageMaker dei job di Processing.

Limitazioni

- Questo modello è adatto ai set di dati ML che includono file interdipendenti. Queste interdipendenze traggono il massimo vantaggio dal macrosharding manuale e dall'esecuzione in parallelo di più processi di elaborazione a istanza singola SageMaker . Per i set di dati in cui tali interdipendenze non esistono, la `ShardedByS3Key` funzionalità di SageMaker Processing potrebbe essere un'alternativa migliore al macrosharding, poiché invia dati suddivisi a più istanze gestite dallo stesso processo di elaborazione. Tuttavia, è possibile implementare la strategia di microsharding di questo pattern in entrambi gli scenari per utilizzare al meglio l'istanza v. CPUs

Versioni del prodotto

- SDK Amazon SageMaker Python versione 2

Architecture

Stack tecnologico Target

- Amazon Simple Storage Service (Amazon S3)
- Amazon SageMaker

Architettura Target

Macrosharding e istanze distribuite EC2

I 10 processi paralleli rappresentati in questa architettura riflettono la struttura del set di dati MIMIC-III. (I processi sono rappresentati da ellissi per semplificare i diagrammi.) Un'architettura simile si applica a qualsiasi set di dati quando si utilizza il macrosharding manuale. Nel caso di MIMIC-III, è possibile utilizzare a proprio vantaggio la struttura grezza del set di dati elaborando ogni cartella del gruppo di pazienti separatamente, con il minimo sforzo. Nel diagramma seguente, il blocco dei gruppi di record appare sulla sinistra (1). Data la natura distribuita dei dati, ha senso condividerli per gruppo di pazienti.

Tuttavia, la suddivisione manuale per gruppo di pazienti significa che è necessario un processo di elaborazione separato per ogni cartella del gruppo di pazienti, come si può vedere nella sezione centrale del diagramma (2), anziché un singolo processo di elaborazione con più istanze. EC2 Poiché i dati di MIMIC-III includono sia file di forme d'onda binarie che file di intestazione basati su testo

corrispondenti e che è richiesta la dipendenza dalla libreria wfdb per l'estrazione dei dati binari, tutti i record per un paziente specifico devono essere resi disponibili sulla stessa istanza. L'unico modo per assicurarsi che sia presente anche il file di intestazione associato a ogni file di forma d'onda binaria è implementare lo sharding manuale per eseguire ogni shard all'interno del proprio processo di elaborazione e specificare `s3_data_distribution_type='FullyReplicated'` quando si definisce l'input del processo di elaborazione. In alternativa, se tutti i dati fossero disponibili in un'unica directory e non esistessero dipendenze tra i file, un'opzione più adatta potrebbe essere quella di avviare un singolo processo di Processing con più istanze specificate. EC2 `s3_data_distribution_type='ShardedByS3Key'` Specificare `ShardedByS3Key` come indicato dal tipo di distribuzione dei dati di Amazon S3 per gestire automaticamente lo SageMaker sharding dei dati tra le istanze.

L'avvio di un processo di elaborazione per ogni cartella è un modo conveniente per preelaborare i dati, perché l'esecuzione simultanea di più istanze consente di risparmiare tempo. Per ulteriori risparmi in termini di costi e tempi, è possibile utilizzare il microsharding all'interno di ciascun processo di elaborazione.

Microsharding e parallel v CPUs

All'interno di ogni processo di elaborazione, i dati raggruppati vengono ulteriormente suddivisi per massimizzare l'utilizzo di tutte le v disponibili CPUs sull'istanza completamente gestita SageMaker . EC2 I blocchi nella sezione centrale del diagramma (2) illustrano ciò che accade all'interno di ciascun processo di elaborazione principale. Il contenuto delle cartelle cliniche dei pazienti viene appiattito e suddiviso equamente in base al numero di v CPUs disponibili sull'istanza. Dopo la divisione del contenuto della cartella, il set di file di dimensioni uguali viene distribuito su tutti i v CPUs per l'elaborazione. Una volta completata l'elaborazione, i risultati di ogni vCPU vengono combinati in un unico file di dati per ogni processo di elaborazione.

Nel codice allegato, questi concetti sono rappresentati nella sezione seguente del `src/feature-engineering-pass1/preprocessing.py` file.

```
def chunks(lst, n):
    """
    Yield successive n-sized chunks from lst.

    :param lst: list of elements to be divided
    :param n: number of elements per chunk
    :type lst: list
    :type n: int
```

```
    :return: generator comprising evenly sized chunks
    :rtype: class 'generator'
    """
    for i in range(0, len(lst), n):
        yield lst[i:i + n]


# Generate list of data files on machine
data_dir = input_dir
d_subs = next(os.walk(os.path.join(data_dir, '.')))[1]
file_list = []
for ds in d_subs:
    file_list.extend(os.listdir(os.path.join(data_dir, ds, '.')))
dat_list = [os.path.join(re.split('_|\.', f)[0].replace('n', ''), f[:-4]) for f in
 file_list if f[-4:] == '.dat']

# Split list of files into sub-lists
cpu_count = multiprocessing.cpu_count()
splits = int(len(dat_list) / cpu_count)
if splits == 0: splits = 1
dat_chunks = list(chunks(dat_list, splits))

# Parallelize processing of sub-lists across CPUs
ws_df_list = Parallel(n_jobs=-1, verbose=0)(delayed(run_process)(dc) for dc in
 dat_chunks)

# Compile and pickle patient group dataframe
ws_df_group = pd.concat(ws_df_list)
ws_df_group = ws_df_group.reset_index().rename(columns={'index': 'signal'})
ws_df_group.to_json(os.path.join(output_dir, group_data_out))
```

Una funzione,`chunks`, viene innanzitutto definita per utilizzare un determinato elenco dividendolo in blocchi di lunghezza di dimensioni uguali n e restituendo questi risultati come generatore. Successivamente, i dati vengono appiattiti tra le cartelle dei pazienti compilando un elenco di tutti i file binari di forme d'onda presenti. Fatto ciò, si ottiene il numero di v CPUs disponibile sull'istanza. EC2 L'elenco dei file binari di forme d'onda viene suddiviso equamente tra questi v CPUs chiamando`chunks`, e quindi ogni sottolista di forme d'onda viene elaborata sulla propria vCPU utilizzando la classe Parallel di joblib. I risultati vengono automaticamente combinati in un unico elenco di dataframe dal processo di elaborazione, che SageMaker quindi li elabora ulteriormente prima di scriverli su Amazon S3 al completamento del processo. In questo esempio, ci sono 10 file scritti su Amazon S3 dai processi di elaborazione (uno per ogni processo).

Una volta completati tutti i processi di elaborazione iniziali, un processo di elaborazione secondario, mostrato nel riquadro a destra del diagramma (3), combina i file di output prodotti da ciascun processo di elaborazione principale e scrive l'output combinato su Amazon S3 (4).

Tools (Strumenti)

Strumenti

- Python — Il codice di esempio usato per questo pattern è Python (versione 3).
- SageMaker Studio: Amazon SageMaker Studio è un ambiente di sviluppo integrato (IDE) basato sul Web per l'apprendimento automatico che ti consente di creare, addestrare, eseguire il debug, distribuire e monitorare i tuoi modelli di machine learning. Puoi eseguire i processi di SageMaker elaborazione utilizzando i notebook Jupyter all'interno di Studio. SageMaker
- SageMaker Elaborazione: Amazon SageMaker Processing offre un modo semplificato per eseguire i carichi di lavoro di elaborazione dei dati. In questo modello, il codice di progettazione delle funzionalità viene implementato su larga scala utilizzando i processi di SageMaker elaborazione.

Codice

Il file.zip allegato fornisce il codice completo per questo pattern. La sezione seguente descrive i passaggi per creare l'architettura per questo pattern. Ogni passaggio è illustrato da un codice di esempio contenuto nell'allegato.

Epiche

Configura il tuo ambiente SageMaker Studio

Operazione	Description	Competenze richieste
Accedi ad Amazon SageMaker Studio.	Effettua l'onboarding su SageMaker Studio nel tuo account AWS seguendo le istruzioni fornite nella SageMaker documentazione di Amazon.	Scienziato dei dati, ingegnere ML
Installa l'utilità wget.	Installa wget se hai effettuat o l'onboarding con una nuova	Scienziato dei dati, ingegnere ML

Operazione	Description	Competenze richieste
	configurazione di SageMaker Studio o se non hai mai usato queste utilità in Studio prima. SageMaker Per installarlo, apri una finestra di terminale nella console di SageMaker Studio ed esegui il seguente comando: `sudo yum install wget`	

Scarica e decomprimi il codice di esempio.	Scarica il `attachments.zip` file nella sezione Allegati. In una finestra di terminale, accedi alla cartella in cui hai scaricato il file ed estraine il contenuto:
	
	`unzip attachment.zip`
	
	Vai alla cartella in cui hai estratto il file.zip ed estrai il contenuto del `Scaled-Processing.zip` file.
	
	`unzip Scaled-Processing.zip`
		Scienziato dei dati, ingegnere ML

Operazione	Description	Competenze richieste
Scarica il set di dati di esempio da physionet.org e caricalo su Amazon S3.	Esegui il notebook `get_data.ipynb` Jupyter all'interno della cartella che contiene i file. `Scaled-Processing` Questo notebook scarica un set di dati MIMIC-III di esempio da physionet.org e lo carica nel bucket di sessione SageMaker Studio in Amazon S3.	Scienziato dei dati, ingegnere ML

Configura il primo script di preelaborazione

Operazione	Description	Competenze richieste

Appiattisci la gerarchia dei file in tutte le sottodirectory.	In set di dati di grandi dimensioni come MIMIC-III , i file sono spesso distribuiti su più sottodirectory anche all'interno di un gruppo principale logico. Lo script deve essere configurato per appiattire tutti i file di gruppo in tutte le sottodirectory, come dimostra il codice seguente.
	`# Generate list of .dat files on machine`
	`data_dir = input_dir`
	`d_subs = next(os.walk(os.path.join(data_dir, '.')))[1]`
	`file_list = []`
	`for ds in d_subs:`
	`  file_list.extend(os.listdir(os.path.`
		Scienziato dei dati, ingegnere ML

Operazione	Description	Competenze richieste

	`join(data_dir, ds, '.')))`
	`dat_list = [os.path.join(re.split('_|\.', f)[0].replace('n', ''), f[:-4]) for f in file_list if f[-4:] == '.dat']`
	
	Note
	I frammenti di codice di esempio di questa epopea provengono dal `src/feature-engineering-pass1/preprocessing.py` file fornito nell'allegato.	

Operazione	Description	Competenze richieste

Dividi i file in sottogruppi in base al numero di vCPU.	I file devono essere divisi in sottogruppi o blocchi di dimensioni uguali, a seconda del numero di v CPUs presenti nell'istanza che esegue lo script. Per questo passaggio, puoi implementare un codice simile al seguente.
	`# Split list of files into sub-lists`
	`cpu_count = multiprocessing.cpu_count()`
	`splits = int(len(dat_list) / cpu_count)`
	`if splits == 0: splits = 1`
	`dat_chunks = list(chunks(dat_list, splits))`
		Scienziato dei dati, ingegnere ML

Operazione	Description	Competenze richieste

Parallelizza l'elaborazione dei sottogruppi su v. CPUs	La logica dello script deve essere configurata per elaborare tutti i sottogruppi in parallelo. A tale scopo, utilizzate la `Parallel` classe e il `delayed` metodo della libreria Joblib come segue.
	`# Parallelize processing of sub-lists across CPUs`
	`ws_df_list = Parallel(n_jobs=-1, verbose=0)(delayed(run_process)(dc) for dc in dat_chunks)`
		Scienziato dei dati, ingegnere ML

Operazione	Description	Competenze richieste

Salva l'output di un singolo gruppo di file su Amazon S3.	Una volta completata l'elaborazione parallela della vCPU, i risultati di ciascuna vCPU devono essere combinati e caricati nel percorso del bucket S3 del gruppo di file. Per questo passaggio, è possibile utilizzare un codice simile al seguente.
	`# Compile and pickle patient group dataframe`
	`ws_df_group = pd.concat(ws_df_list)`
	`ws_df_group = ws_df_group.reset_index().rename(columns={'index': 'signal'})`
	`ws_df_group.to_json(os.path.join(output_dir, group_data_out))`
		Scienziato dei dati, ingegnere ML

Configura il secondo script di preelaborazione

Operazione	Description	Competenze richieste
Combina i file di dati prodotti in tutti i processi di elaborazione che hanno eseguito il primo script.	Lo script precedente genera un singolo file per ogni processo di SageMaker elaborazione che elabora un gruppo di file dal set di dati. Successivamente, è necessario combinare questi file di output in un unico	Scienziato dei dati, ingegnere ML

Operazione	Description	Competenze richieste

	oggetto e scrivere un singolo set di dati di output su Amazon S3. Ciò è dimostrato nel `src/feature-engineering-pass1p5/preprocessing.py` file, fornito nell'allegato, come segue.
	
	`def write_parquet(wavs_df, path):`
	`"""`
	`Write waveform summary dataframe to S3 in parquet format.`
	
	`:param wavs_df: waveform summary dataframe`
	`:param path: S3 directory prefix`
	`:type wavs_df: pandas dataframe`
	`:type path: str`
	`:return: None`
	`"""`
	`extra_args = {"ServerSideEncryption": "aws:kms"}`
	`wr.s3.to_parquet(`
	`df=wavs_df,`
	`path=path,`
	`compression='snappy',`
	`s3_additional_kwargs=extra_args)`
	
	`def combine_data():`
	`"""`	

Operazione	Description	Competenze richieste
	Get combined data and write to parquet. :return: waveform summary dataframe :rtype: pandas dataframe """ wavs_df = get_data() wavs_df = normalize_signal_names(wavs_df) write_parquet(wavs_df, "s3://{}/{}/{}".format(bucket_xform, dataset_prefix, pass1p5out_data)) return wavs_df wavs_df = combine_data()	

Esegui processi di elaborazione

Operazione	Description	Competenze richieste
Esegui il primo processo di elaborazione.	Per eseguire il macroshar ding, esegui un processo di elaborazione separato per ogni gruppo di file. Il microshar ding viene eseguito all'interno di ogni processo di elaborazi one, poiché ogni lavoro esegue il primo script. Il codice	Scienziato dei dati, ingegnere ML

Operazione	Description	Competenze richieste

	seguente mostra come avviare un processo di elaborazione per ogni directory del gruppo di file nel seguente frammento (incluso in). `notebooks/FeatExtract_Pass1.ipynb`
	
	`pat_groups = list(range(30,40))`
	`ts = str(int(time.time()))`
	
	`for group in pat_groups:`
	`  sklearn_processor = SKLearnProcessor(framework_version='0.20.0',`
	
	`  role=role,`
	
	`  instance_type='ml.m5.4xlarge',`
	
	`  instance_count=1,`
	
	`  volume_size_in_gb=5)`
	`  sklearn_processor.run(`
	`  code='../src/feature-engineering-pass1/preprocessing.py',`
	`  job_name='-'.join(['scaled-processing-p1',`
	`  str(group), ts]),`	

Operazione	Description	Competenze richieste
	`arguments=["input_path", "/opt/ml/processing/input", "output_path", "/opt/ml/processing/output", "group_data_out", "ws_df_group.json"], inputs= [ProcessingInput(source=f's3://{sess.default_bucket()}/data_inputs/{group}', destination='/opt/ml/processing/input', s3_data_distribution_type='FullyReplicated')], outputs= [ProcessingOutput(source='/opt/ml/processing/output', destination=f's3://{sess.default_bucket()}/data_outputs/{group}')],`	

Operazione	Description	Competenze richieste
	`wait=False` `)`	

Operazione	Description	Competenze richieste

Esegui il secondo processo di elaborazione.	Per combinare gli output generati dal primo set di processi di elaborazione ed eseguire eventuali calcoli aggiuntivi per la preelabor azione, si esegue il secondo script utilizzando un singolo SageMaker processo di elaborazione. Il codice seguente lo dimostra (incluso in). `notebooks/FeatExtract_Pass1p5.ipynb`
	`ts = str(int(time.time()))`
	`bucket = sess.default_bucket()`
	
	`sklearn_processor = SKLearnProcessor(framework_version='0.20.0',`
	
	`role=role,`
	
	`instance_type='ml.t3.2xlarge',`
	
	`instance_count=1,`
	
	`volume_size_in_gb=5)`
	`sklearn_processor.run(`
	`code='../src/feature-engineering-pass1p5/preprocessing.py',`
		Scienziato dei dati, ingegnere ML

Operazione	Description	Competenze richieste
	`job_name='-'.join(['scaled-processing', 'p1p5', ts]), arguments=['bucket', bucket, 'pass1out_prefix', 'data_outputs', 'pass1out_data', 'ws_df_group.json', 'pass1p5out_data', 'waveform_summary.parquet', 'statsdata_name', 'signal_stats.csv'], wait=True )`	

Risorse correlate

- Effettua l'onboarding su Amazon SageMaker Studio utilizzando Quick Start (SageMaker documentazione)
- Dati di processo (SageMaker documentazione)
- Elaborazione dei dati con scikit-learn (documentazione) SageMaker
- Documentazione Joblib.parallel
- Moody, B., Moody, G., Villarroel, M., Clifford, G.D. e Silva, I. (2020). Database delle forme d'onda MIMIC-III (versione 1.0). PhysioNet.
- Johnson, A.E.W., Pollard, TJ, Shen, L., Lehman, L.H., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L.A. e Mark, R.G. (2016). MIMIC-III, un database di terapia intensiva accessibile gratuitamente. Dati scientifici, 3, 160035.
- Licenza del database MIMIC-III Waveform

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Visualizza i risultati AI/ML del modello utilizzando Flask e AWS Elastic Beanstalk

Chris Caudill e Durga Sury, Amazon Web Services

Riepilogo

La visualizzazione dei risultati dei servizi di intelligenza artificiale e machine learning (AI/ML) spesso richiede chiamate API complesse che devono essere personalizzate dai tuoi sviluppatori e ingegneri. Questo può essere uno svantaggio se gli analisti vogliono esplorare rapidamente un nuovo set di dati.

È possibile migliorare l'accessibilità dei servizi e fornire una forma più interattiva di analisi dei dati utilizzando un'interfaccia utente (UI) basata sul Web che consente agli utenti di caricare i propri dati e visualizzare i risultati del modello in una dashboard.

Questo modello utilizza Flask e Plotly per integrare Amazon Comprehend con un'applicazione web personalizzata e visualizzare sentimenti ed entità a partire dai dati forniti dagli utenti. Il modello fornisce anche i passaggi per distribuire un'applicazione utilizzando AWS Elastic Beanstalk. Puoi adattare l'applicazione utilizzando i servizi di intelligenza artificiale di Amazon Web Services (AWS) o con un modello addestrato personalizzato ospitato su un endpoint (ad esempio, un SageMaker endpoint Amazon).

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- AWS Command Line Interface (AWS CLI), installata e configurata sul computer locale. Per ulteriori informazioni su questo argomento, consulta le nozioni di base sulla configurazione nella documentazione dell'interfaccia a riga di comando di AWS. Puoi anche utilizzare un ambiente di sviluppo integrato (IDE) AWS Cloud9; per ulteriori informazioni su questo argomento, consulta il tutorial di Python per AWS Cloud9 e l'anteprima delle applicazioni in esecuzione nell'IDE AWS Cloud9 nella documentazione di AWS Cloud9.

  Avviso: non è più disponibile per i nuovi clienti. AWS Cloud9 I clienti esistenti di AWS Cloud9 possono continuare a utilizzare il servizio normalmente. Ulteriori informazioni
- Comprensione del framework di applicazioni web di Flask. Per ulteriori informazioni su Flask, consulta il Quickstart nella documentazione di Flask.

- Python versione 3.6 o successiva, installato e configurato. Puoi installare Python seguendo le istruzioni contenute in Configurazione dell'ambiente di sviluppo Python nella documentazione di AWS Elastic Beanstalk.
- Elastic Beanstalk Command Line Interface (EB CLI), installata e configurata. Per ulteriori informazioni su questo argomento, consulta Installare l'EB CLI e Configurare l'EB CLI dalla documentazione di AWS Elastic Beanstalk.

Limitazioni

- L'applicazione Flask di questo pattern è progettata per funzionare con file.csv che utilizzano una singola colonna di testo e sono limitati a 200 righe. Il codice dell'applicazione può essere adattato per gestire altri tipi di file e volumi di dati.
- L'applicazione non considera la conservazione dei dati e continua ad aggregare i file utente caricati fino a quando non vengono eliminati manualmente. Puoi integrare l'applicazione con Amazon Simple Storage Service (Amazon S3) per lo storage persistente di oggetti o utilizzare un database come Amazon DynamoDB per lo storage di chiave-valore senza server.
- L'applicazione prende in considerazione solo i documenti in lingua inglese. Tuttavia, puoi utilizzare Amazon Comprehend per rilevare la lingua principale di un documento. Per ulteriori informazioni sulle lingue supportate per ogni azione, consulta il riferimento all'API nella documentazione di Amazon Comprehend.
- Un elenco di risoluzione dei problemi che contiene gli errori più comuni e le relative soluzioni è disponibile nella sezione Informazioni aggiuntive.

Architecture

Architettura dell'applicazione Flask

Flask è un framework leggero per lo sviluppo di applicazioni web in Python. È progettato per combinare la potente elaborazione dei dati di Python con una ricca interfaccia utente web. L'applicazione Flask del pattern mostra come creare un'applicazione Web che consenta agli utenti di caricare dati, inviarli ad Amazon Comprehend per l'inferenza e quindi visualizzare i risultati. L'applicazione ha la seguente struttura:

- `static`— Contiene tutti i file statici che supportano l'interfaccia utente Web (ad esempio JavaScript, CSS e immagini)
- `templates`— Contiene tutte le pagine HTML dell'applicazione

- `userData`— Memorizza i dati utente caricati
- `application.py`— Il file dell'applicazione Flask
- `comprehend_helper.py`— Funzioni per effettuare chiamate API verso Amazon Comprehend
- `config.py`— Il file di configurazione dell'applicazione
- `requirements.txt`— Le dipendenze Python richieste dall'applicazione

Lo `application.py` script contiene le funzionalità principali dell'applicazione Web, che consiste in quattro percorsi Flask. Il diagramma seguente mostra questi percorsi Flask.

- `/`è la radice dell'applicazione e indirizza gli utenti alla `upload.html` pagina (memorizzata nella `templates` directory).
- `/saveFile`è una route che viene richiamata dopo che un utente carica un file. Questo percorso riceve una `POST` richiesta tramite un modulo HTML, che contiene il file caricato dall'utente. Il file viene salvato nella `userData` directory e il percorso reindirizza gli utenti al `/dashboard` percorso.
- `/dashboard`invia gli utenti alla pagina. `dashboard.html` All'interno del codice HTML di questa pagina, esegue il JavaScript codice `static/js/core.js` che legge i dati dal `/data` percorso e quindi crea visualizzazioni per la pagina.
- `/data`è un'API JSON che presenta i dati da visualizzare nella dashboard. Questo percorso legge i dati forniti dall'utente e utilizza le funzioni `comprehend_helper.py` per inviare i dati utente ad Amazon Comprehend per l'analisi del sentiment e il riconoscimento delle entità nominate (NER). La risposta di Amazon Comprehend viene formattata e restituita come oggetto JSON.

Architettura di distribuzione

Considerazioni di progettazione

Per ulteriori informazioni sulle considerazioni di progettazione per le applicazioni distribuite utilizzando Elastic Beanstalk sul cloud AWS, consulta la documentazione di AWS Elastic Beanstalk.

Stack tecnologico

- Amazon Comprehend
- Elastic Beanstalk

- Flask

Automazione e scalabilità

Le implementazioni di Elastic Beanstalk vengono configurate automaticamente con sistemi di bilanciamento del carico e gruppi di auto scaling. Per ulteriori opzioni di configurazione, consulta Configurazione degli ambienti Elastic Beanstalk nella documentazione di AWS Elastic Beanstalk.

Tools (Strumenti)

- AWS Command Line Interface (AWS CLI) è uno strumento unificato che fornisce un'interfaccia coerente per interagire con tutte le parti di AWS.
- Amazon Comprehend utilizza l'elaborazione del linguaggio naturale (NLP) per estrarre informazioni dettagliate sul contenuto dei documenti senza richiedere una preelaborazione speciale.
- AWS Elastic Beanstalk ti aiuta a distribuire e gestire rapidamente le applicazioni nel cloud AWS senza dover conoscere l'infrastruttura che esegue tali applicazioni.
- Elastic Beanstalk CLI (EB CLI) è un'interfaccia a riga di comando per AWS Elastic Beanstalk che fornisce comandi interattivi per semplificare la creazione, l'aggiornamento e il monitoraggio di ambienti da un repository locale.
- Il framework Flask esegue l'elaborazione dei dati e le chiamate API utilizzando Python e offre una visualizzazione web interattiva con Plotly.

Codice

Il codice per questo modello è disponibile nei risultati del AI/ML modello GitHub Visualize utilizzando Flask e il repository AWS Elastic Beanstalk.

Epiche

Configura l'applicazione Flask

Operazione	Description	Competenze richieste
Clona il GitHub repository.	Estrai il codice dell'appl icazione dai risultati del AI/ ML modello GitHub Visualize utilizzando Flask e il repositor	Developer

Operazione	Description	Competenze richieste
	y AWS Elastic Beanstalk eseguendo il seguente comando: `git clone git@github.com:aws-samples/aws-comprehend-elasticbeanstalk-for-flask.git` **Note** Assicurati di configurare le tue chiavi SSH con. GitHub	

Installa i moduli Python.	Dopo aver clonato il repository, viene creata una nuova `aws-comprehend-elasticbeanstalk-for-flask` directory locale. In quella directory, il `requirements.txt` file contiene i moduli e le versioni di Python che eseguono l'applicazione. Utilizzate i seguenti comandi per installare i moduli:
	
	`cd aws-comprehend-elasticbeanstalk-for-flask`
	
	`pip install -r requirements.txt`
		Sviluppatore Python

Operazione	Description	Competenze richieste

Prova l'applicazione localmente.	Avvia il server Flask eseguendo il seguente comando:
	
	`python application.py`
	
	Ciò restituisce informazioni sul server in esecuzione. Dovresti essere in grado di accedere all'applicazione aprendo un browser e visitando http://localhost:5000
	
	Note
	Se esegui l'applicazione in un IDE AWS Cloud9, devi sostituire `application.run()` il comando nel file con `application.py` la seguente riga:
	
	`application.run(host=os.getenv('IP', '0.0.0.0'),port=int(os.getenv('PORT', 8080)))`
	
	È necessario annullare questa modifica prima della distribuzione.
		Sviluppatore Python

Distribuisci l'applicazione su Elastic Beanstalk

Operazione	Description	Competenze richieste
Avvia l'applicazione Elastic Beanstalk.	Per avviare il progetto come applicazione Elastic Beanstalk , esegui il seguente comando dalla directory principale dell'applicazione: `eb init -p python-3.6 comprehend_flask --region us-east-1` ⚠ Important • `comprehend_flask` è il nome dell'applicazione Elastic Beanstalk e può essere modificato in base alle proprie esigenze. • Puoi sostituire la regione AWS con una regione a tua scelta. La regione predefinita in AWS CLI viene utilizzata se non si specifica una regione. • L'applicazione è stata creata con Python versione 3.6. Potresti riscontrare errori se usi altre versioni di Python. Esegui il `eb init -i` comando per ulteriori opzioni	Architetto, sviluppatore

Operazione	Description	Competenze richieste
	di configurazione della distribuzione.	
Implementa l'ambiente Elastic Beanstalk.	Esegui il comando seguente dalla directory principale dell'applicazione: `eb create comprehend-flask-env` **Note** `comprehend-flask-env` è il nome dell'ambiente Elastic Beanstalk e può essere modificato in base alle proprie esigenze. Il nome può contenere solo lettere, numeri e trattini.	Architetto, sviluppatore

Operazione	Description	Competenze richieste
Autorizza la tua distribuzione all'uso di Amazon Comprehend.	Sebbene la tua applicazione possa essere stata distribuita correttamente, dovresti anche fornire alla distribuzione l'accesso ad Amazon Comprehend. `ComprehendFullAccess` è una policy gestita da AWS che fornisce all'applicazione distribuita le autorizzazioni per effettuare chiamate API verso Amazon Comprehend. Allega la `ComprehendFullAccess` policy a `aws-elasticbeanstalk-ec2-role` (questo ruolo viene creato automaticamente per le istanze Amazon Elastic Compute Cloud (Amazon EC2) della tua distribuzione) eseguendo il seguente comando: `aws iam attach-role-policy --policy-arn arn:aws:iam::aws:policy/ComprehendFullAccess --role-name aws-elasticbeanstalk-ec2-role`	Sviluppatore, architetto della sicurezza

Operazione	Description	Competenze richieste
	Important `aws-elasticbeanstalk-ec2-role` viene creato quando l'applicazione viene distribuita. È necessario completare il processo di distribuzione prima di poter allegare la policy di AWS Identity and Access Management (IAM).	
Visita la tua applicazione distribuita.	Dopo che l'applicazione è stata distribuita correttamente, puoi visitarla eseguendo il `eb open` comando. Puoi anche eseguire il `eb status` comando per ricevere dettagli sulla distribuzione. L'URL di distribuzione è elencato sotto`CNAME`.	Architetto, sviluppatore

(Facoltativo) Personalizza l'applicazione in base al tuo modello ML

Operazione	Description	Competenze richieste
Autorizza Elastic Beanstalk ad accedere al nuovo modello.	Assicurati che Elastic Beanstalk disponga delle autorizzazioni di accesso	Sviluppatore, architetto della sicurezza

Operazione	Description	Competenze richieste
	necessarie per il tuo nuovo endpoint modello. Ad esempio, se utilizzi un SageMaker endpoint Amazon, la tua distribuzione deve disporre dell'autorizzazione per richiamare l'endpoint. Per ulteriori informazioni a riguardo, InvokeEndpointconsulta la SageMaker documentazione di Amazon.	

Operazione	Description	Competenze richieste
Invia i dati dell'utente a un nuovo modello.	Per modificare il modello ML sottostante in questa applicazione, è necessario modificare i seguenti file: • `comprehend_helper.py` — Si tratta dello script Python che si connette ad Amazon Comprehend, elabora la risposta e restituisce il risultato finale all'applicazione. In questo script, puoi indirizzare i dati a un altro servizio di intelligenza artificiale sul cloud AWS oppure puoi inviare i dati a un endpoint modello personalizzato. Ti consigliamo di formattare anche i risultati in questo script per la separazione logica e la riutilizzabilità di questo pattern. • `application.py` — Se si modifica il nome dello `comprehend_helper.py` script o delle funzioni, è necessario aggiornare `application.py` lo script dell'applicazione per riflettere tali modifiche.	Data Scientist

Operazione	Description	Competenze richieste
Aggiorna le visualizzazioni del pannello di controllo.	In genere, incorporare un nuovo modello ML significa che le visualizzazioni devono essere aggiornate per rifletter e i nuovi risultati. Queste modifiche vengono apportate nei seguenti file: • `templates/dashboard.html` — L'applicazione predefinita tiene conto solo di due visualizzazioni di base. L'intero layout della pagina può essere modificato in questo file. • `static/js/core.js` — Questo script acquisisce l'output formattato del `/data` percorso del server Flask e utilizza Plotly per creare visualizzazioni. Puoi aggiungere o aggiornare i grafici della pagina.	Sviluppatore web

(Facoltativo) Distribuisci l'applicazione aggiornata

Operazione	Description	Competenze richieste
Aggiorna il file dei requisiti dell'applicazione.	Prima di inviare modifiche a Elastic Beanstalk, `requirements.txt` aggiorna il file in modo che rifletta eventuali nuovi moduli Python eseguendo il seguente	Sviluppatore Python

Operazione	Description	Competenze richieste
	comando nella directory principale dell'applicazione: `pip freeze > requirements.txt`	
Ridistribuisci l'ambiente Elastic Beanstalk.	Per assicurarti che le modifiche all'applicazione si riflettano nella distribuzione di Elastic Beanstalk, vai alla directory principale dell'applicazione ed esegui il comando seguente: `eb deploy` Questo invia la versione più recente del codice dell'applicazione alla distribuzione esistente di Elastic Beanstalk.	Amministratore di sistema, Architetto

Risorse correlate

- Chiama un endpoint SageMaker modello Amazon utilizzando Amazon API Gateway e AWS Lambda
- Distribuzione di un'applicazione Flask su Elastic Beanstalk
- Riferimento ai comandi CLI EB
- Configurazione dell'ambiente di sviluppo Python

Informazioni aggiuntive

Elenco di risoluzione dei problemi

Di seguito sono riportati sei errori comuni e le relative soluzioni.

Errore 1

```
Unable to assume role "arn:aws:iam::xxxxxxxxxx:role/aws-elasticbeanstalk-ec2-role".
 Verify that the role exists and is configured correctly.
```

Soluzione: se questo errore si verifica durante l'esecuzione`eb create`, crea un'applicazione di esempio sulla console Elastic Beanstalk per creare il profilo di istanza predefinito. Per ulteriori informazioni su questo argomento, consulta Creazione di un ambiente Elastic Beanstalk nella documentazione di AWS Elastic Beanstalk.

Errore 2

```
Your WSGIPath refers to a file that does not exist.
```

Soluzione: questo errore si verifica nei registri di distribuzione perché Elastic Beanstalk prevede che il codice Flask venga denominato. `application.py` Se hai scelto un nome diverso, eseguilo `eb config` e modificalo WSGIPath come mostrato nel seguente esempio di codice:

```
aws:elasticbeanstalk:container:python:
     NumProcesses: '1'
     NumThreads: '15'
     StaticFiles: /static/=static/
     WSGIPath: application.py
```

Assicurati di sostituirlo `application.py` con il nome del file.

Puoi anche sfruttare Gunicorn e un Procfile. Per ulteriori informazioni su questo approccio, consulta Configurazione del server WSGI con un Procfile nella documentazione di AWS Elastic Beanstalk.

Errore 3

```
Target WSGI script '/opt/python/current/app/application.py' does not contain WSGI
 application 'application'.
```

Soluzione: Elastic Beanstalk si aspetta che la variabile che rappresenta l'applicazione Flask venga denominata. `application` Assicurati che il `application.py` file utilizzi `application` come nome della variabile:

```
application = Flask(__name__)
```

Errore 4

```
The EB CLI cannot find your SSH key file for keyname
```

Soluzione: utilizza l'EB CLI per specificare quale coppia di chiavi utilizzare o per creare una coppia di chiavi per le istanze EC2 della distribuzione. Per risolvere l'errore, esegui `eb init -i` e una delle opzioni chiederà:

```
Do you want to set up SSH for your instances?
```

YRispondi con per creare una coppia di chiavi o specificare una coppia di chiavi esistente.

Errore 5

Ho aggiornato il codice e l'ho ridistribuito, ma la mia distribuzione non riflette le mie modifiche.

Soluzione: se utilizzi un repository Git con la tua distribuzione, assicurati di aggiungere e confermare le modifiche prima di ridistribuirle.

Errore 6

Stai visualizzando l'anteprima dell'applicazione Flask da un IDE AWS Cloud9 e riscontri degli errori.

Soluzione: per ulteriori informazioni su questo argomento, consulta Anteprima delle applicazioni in esecuzione nell'IDE AWS Cloud9 nella documentazione di AWS Cloud9.

Elaborazione del linguaggio naturale con Amazon Comprehend

Scegliendo di utilizzare Amazon Comprehend, puoi rilevare entità personalizzate in singoli documenti di testo eseguendo analisi in tempo reale o processi batch asincroni. Amazon Comprehend consente inoltre di addestrare modelli personalizzati di riconoscimento delle entità e classificazione del testo che possono essere utilizzati in tempo reale creando un endpoint.

Questo modello utilizza processi batch asincroni per rilevare sentimenti ed entità da un file di input che contiene più documenti. L'applicazione di esempio fornita da questo modello è progettata per consentire agli utenti di caricare un file.csv contenente una singola colonna con un documento di testo per riga. Il `comprehend_helper.py` file nei risultati del AI/ML modello GitHub Visualize utilizzando Flask e il repository AWS Elastic Beanstalk legge il file di input e invia l'input ad Amazon Comprehend per l'elaborazione.

BatchDetectEntities

Amazon Comprehend esamina il testo di un batch di documenti alla ricerca di entità denominate e restituisce l'entità, l'ubicazione, il tipo di entità rilevati e un punteggio che indica il livello di fiducia di Amazon Comprehend. È possibile inviare un massimo di 25 documenti in una chiamata API, con ogni documento di dimensioni inferiori a 5.000 byte. Puoi filtrare i risultati per mostrare solo determinate entità in base al caso d'uso. Ad esempio, è possibile ignorare il tipo di `'quantity'` entità e impostare un punteggio di soglia per l'entità rilevata (ad esempio, 0,75). Ti consigliamo di esaminare i risultati per il tuo caso d'uso specifico prima di scegliere un valore di soglia. Per ulteriori informazioni su questo argomento, consulta la BatchDetectEntitiesdocumentazione di Amazon Comprehend.

BatchDetectSentiment

Amazon Comprehend ispeziona un batch di documenti in entrata e restituisce il sentimento prevalente per ogni documento (,, o). `POSITIVE` `NEUTRAL` `MIXED` `NEGATIVE` È possibile inviare un massimo di 25 documenti in una chiamata API, con ogni documento di dimensioni inferiori a 5.000 byte. L'analisi del sentimento è semplice e puoi scegliere il sentimento con il punteggio più alto da visualizzare nei risultati finali. Per ulteriori informazioni su questo argomento, consulta la BatchDetectSentimentdocumentazione di Amazon Comprehend.

Gestione della configurazione di Flask

I server Flask utilizzano una serie di variabili di configurazione per controllare il funzionamento del server. Queste variabili possono contenere output di debug, token di sessione o altre impostazioni dell'applicazione. È inoltre possibile definire variabili personalizzate a cui è possibile accedere mentre l'applicazione è in esecuzione. Esistono diversi approcci per l'impostazione delle variabili di configurazione.

In questo modello, la configurazione è definita `config.py` ed ereditata all'interno`application.py`.

- > **Note**
  >
  > `config.py`contiene le variabili di configurazione impostate all'avvio dell'applicazione. In questa applicazione, viene definita una `DEBUG` variabile per indicare all'applicazione di eseguire il server in modalità di debug. : la modalità di debug non deve essere utilizzata quando si esegue un'applicazione in un ambiente di produzione. `UPLOAD_FOLDER`è una

variabile personalizzata definita per essere referenziata più avanti nell'applicazione e che indica dove devono essere archiviati i dati utente caricati.

- `application.py`avvia l'applicazione Flask ed eredita le impostazioni di configurazione definite in. `config.py` Questa operazione viene eseguita dal seguente codice:

```
application = Flask(__name__)
application.config.from_pyfile('config.py')
```

Altri modelli

- Accelera MLOps con Backstage e modelli Amazon SageMaker AI self-service
- Automatizza le operazioni AWS dell'infrastruttura utilizzando Amazon Bedrock
- Implementa sistemi agentici su Amazon Bedrock con il framework CrewAI utilizzando Terraform
- Implementa una ChatOps soluzione per gestire i risultati della scansione SAST utilizzando Amazon Q Developer nelle applicazioni di chat, azioni personalizzate e CloudFormation
- Genera approfondimenti sui dati utilizzando Modernizzazione del mainframe AWS Amazon Q in Quick Sight
- Genera approfondimenti sui z/OS dati Db2 utilizzando Modernizzazione del mainframe AWS Amazon Q in Quick Sight
- Offri alle istanze di SageMaker notebook l'accesso temporaneo a un CodeCommit repository in un altro account AWS
- Gestisci AWS Organizations le policy come codice utilizzando AWS CodePipeline Amazon Bedrock
- Modernizza l'applicazione CardDemo mainframe utilizzando AWS Transform
- Modernizza e implementa le applicazioni mainframe utilizzando e Terraform AWS Transform
- Esegui analisi avanzate con Amazon Redshift ML
- Semplifica la gestione EC2 della conformità di Amazon con agenti Amazon Bedrock e AWS Config
- Semplifica lo sviluppo e la distribuzione di bot Amazon Lex utilizzando un flusso di lavoro automatizzato
- Risolvi i problemi relativi agli stati AWS Step Functions utilizzando Amazon Bedrock

Analisi

Argomenti

- Analizza i dati di Amazon Redshift in Microsoft SQL Server Analysis Services
- Analizza e visualizza dati JSON annidati con Amazon Athena e Amazon Quick Sight
- Automatizza l'inserimento dei dati da Amazon S3 AWS Data Exchange
- Automatizza l'applicazione della crittografia in AWS Glue utilizzando un modello AWS CloudFormation
- Crea una pipeline di dati per importare, trasformare e analizzare i dati di Google Analytics utilizzando il AWS DataOps Development Kit
- Crea una pipeline di elaborazione video utilizzando Amazon Kinesis Video Streams e AWS Fargate
- Crea una pipeline di servizi ETL per caricare i dati in modo incrementale da Amazon S3 ad Amazon Redshift utilizzando AWS Glue
- Crea una rete dati aziendale con Amazon DataZone e AWS CDK AWS CloudFormation
- Calcola il valore a rischio (VaR) utilizzando i servizi AWS
- Configura l'accesso tra account a un catalogo di dati AWS Glue condiviso utilizzando Amazon Athena
- Convertire la funzionalità temporale Teradata NORMALIZE in Amazon Redshift SQL
- Convertire la funzionalità Teradata RESET WHEN in Amazon Redshift SQL
- Implementa e gestisci un data lake serverless sul cloud AWS utilizzando l'infrastruttura come codice
- Applica l'etichettatura dei cluster Amazon EMR al momento del lancio
- Assicurati che la registrazione di Amazon EMR su Amazon S3 sia abilitata al momento del lancio
- Genera dati di test utilizzando un job AWS Glue e Python
- Inserimento conveniente di dati IoT direttamente in Amazon S3 con AWS IoT Greengrass
- Avvia un job Spark in un cluster EMR transitorio utilizzando una funzione Lambda
- Esegui la migrazione dei carichi di lavoro Apache Cassandra su Amazon Keyspaces utilizzando AWS Glue
- Esegui la migrazione di Oracle Business Intelligence 12c al cloud AWS dai server locali
- Migrazione di uno stack ELK su Elastic Cloud su AWS
- Migrare i dati verso il Cloud AWS utilizzando Starburst

- Ottimizza l'ingestione ETL delle dimensioni dei file di input su AWS
- Orchestra una pipeline ETL con convalida, trasformazione e partizionamento utilizzando AWS Step Functions
- Esegui analisi avanzate con Amazon Redshift ML
- Interroga le tabelle Amazon DynamoDB con SQL utilizzando Amazon Athena
- Accedi, esegui query e unisciti a tabelle Amazon DynamoDB utilizzando Athena
- Imposta uno spazio dati minimo praticabile per condividere i dati tra le organizzazioni
- Imposta l'ordinamento specifico della lingua per i risultati delle query di Amazon Redshift utilizzando un UDF scalare in Python
- Sottoscrivi una funzione Lambda alle notifiche di eventi dai bucket S3 in diverse regioni AWS
- Tre tipi di job ETL di AWS Glue per la conversione dei dati in Apache Parquet
- Visualizza i log di controllo di Amazon Redshift utilizzando Amazon Athena e Amazon QuickSight
- Visualizza i report sulle credenziali IAM per tutti gli account AWS utilizzando Amazon Quick Sight
- Altri modelli

Analizza i dati di Amazon Redshift in Microsoft SQL Server Analysis Services

Sunil Vora, Amazon Web Services

Riepilogo

Questo modello descrive come connettere e analizzare i dati di Amazon Redshift in Microsoft SQL Server Analysis Services, utilizzando il provider Intellisoft OLE DB o il provider CData ADO.NET per l'accesso al database.

Amazon Redshift è un servizio di data warehouse nel cloud in scala petabyte interamente gestito. SQL Server Analysis Services è uno strumento di elaborazione analitica online (OLAP) che puoi utilizzare per analizzare i dati provenienti da data mart e data warehouse come Amazon Redshift. Puoi utilizzare SQL Server Analysis Services per creare cubi OLAP dai tuoi dati per un'analisi rapida e avanzata dei dati.

Prerequisiti e limitazioni

Ipotesi

- Questo modello descrive come configurare SQL Server Analysis Services e Intellisoft OLE DB Provider o CData ADO.NET Provider per Amazon Redshift su un'istanza Amazon Elastic Compute Cloud (Amazon). EC2 In alternativa, puoi installarli entrambi su un host nel tuo data center aziendale.

Prerequisiti

- Un account AWS attivo
- Un cluster Amazon Redshift con credenziali

Architecture

Stack tecnologico di origine

- Un cluster Amazon Redshift

Stack tecnologico Target

- Servizi di analisi Microsoft SQL Server

Architettura di origine e destinazione

Tools (Strumenti)

- Microsoft Visual Studio 2019 (edizione comunitaria)
- Intellisoft OLE DB Provider per Amazon Redshift (versione di prova) CData o ADO.NET Provider per Amazon Redshift (versione di prova)

Epiche

Analizza le tabelle

Operazione	Description	Competenze richieste
Analizza le tabelle e i dati da importare.	Identifica le tabelle Amazon Redshift da importare e le relative dimensioni.	DBA

Configura l' EC2 istanza e installa gli strumenti

Operazione	Description	Competenze richieste
Configura un' EC2 istanza.	Nel tuo account AWS, crea un' EC2 istanza in una sottorete pubblica o privata.	Amministratore di sistema
Installa strumenti per l'accesso al database.	Scarica e installa Intellisoft OLE DB Provider per Amazon Redshift (CData o ADO.NET Provider per Amazon Redshift) .	Amministratore di sistema

Operazione	Description	Competenze richieste
Installa Visual Studio.	Scarica e installa Visual Studio 2019 (Community Edition).	Amministratore di sistema
Installa le estensioni.	Installa l'estensione Microsoft Analysis Services Projects in Visual Studio.	Amministratore di sistema
Crea un progetto.	Crea un nuovo progetto di modello tabulare in Visual Studio per archiviare i dati di Amazon Redshift. In Visual Studio, scegli l'opzione Analysis Services Tabular Project durante la creazione del progetto.	DBA

Crea sorgenti di dati e importa tabelle

Operazione	Description	Competenze richieste
Crea un'origine dati Amazon Redshift.	Crea un'origine dati Amazon Redshift utilizzando Intellisoft OLE DB Provider per Amazon Redshift (o ADO.NET CData Provider per Amazon Redshift) e le tue credenziali Amazon Redshift.	Amazon Redshift, DBA
Importa tabelle.	Seleziona e importa tabelle e viste da Amazon Redshift nel tuo progetto SQL Server Analysis Services.	Amazon Redshift, DBA

Pulizia dopo la migrazione

Operazione	Description	Competenze richieste
Eliminare l' EC2 istanza.	Elimina l' EC2 istanza che hai avviato in precedenza.	Amministratore di sistema

Risorse correlate

- Amazon Redshift (documentazione AWS)
- Installare SQL Server Analysis Services (documentazione Microsoft)
- Tabular Model Designer (documentazione Microsoft)
- Panoramica dei cubi OLAP per analisi avanzate (documentazione Microsoft)
- Microsoft Visual Studio 2019 (edizione comunitaria)
- Intellisoft OLE DB Provider per Amazon Redshift (versione di prova)
- CData Provider ADO.NET per Amazon Redshift (versione di prova)

Analizza e visualizza dati JSON annidati con Amazon Athena e Amazon Quick Sight

Anoop Singh, Amazon Web Services

Riepilogo

Questo modello spiega come tradurre una struttura di dati annidata in formato JSON in una vista tabulare utilizzando Amazon Athena e quindi visualizzare i dati in Amazon Quick Sight.

Puoi utilizzare dati in formato JSON per feed di dati basati su API provenienti da sistemi operativi per creare prodotti di dati. Questi dati possono anche aiutarti a comprendere meglio i tuoi clienti e le loro interazioni con i tuoi prodotti, in modo da personalizzare le esperienze degli utenti e prevedere i risultati.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un file JSON che rappresenta una struttura di dati annidata (questo modello fornisce un file di esempio)

Limitazioni:

- Le funzionalità JSON si integrano bene con le funzioni orientate a SQL esistenti in Athena. Tuttavia, non sono compatibili con ANSI SQL e si prevede che il file JSON contenga ogni record su una riga separata. Potrebbe essere necessario utilizzare la `ignore.malformed.json` proprietà in Athena per indicare se i record JSON non validi devono essere trasformati in caratteri nulli o generare errori. Per ulteriori informazioni, consulta Best practice per la lettura dei dati JSON nella documentazione di Athena.
- Questo modello considera solo piccole e semplici quantità di dati in formato JSON. Se desideri utilizzare questi concetti su larga scala, prendi in considerazione l'applicazione del partizionamento dei dati e il consolidamento dei dati in file più grandi.

Architecture

Il diagramma seguente mostra l'architettura e il flusso di lavoro per questo modello. Le strutture di dati annidate sono archiviate in Amazon Simple Storage Service (Amazon S3) in formato JSON. In Athena, i dati JSON vengono mappati su una struttura dati Athena. Quindi crei una vista per analizzare i dati e visualizzare la struttura dei dati in Quick Sight.

Tools (Strumenti)

Servizi AWS

- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati. Questo modello utilizza Amazon S3 per archiviare il file JSON.
- Amazon Athena è un servizio di query interattivo che ti aiuta ad analizzare i dati direttamente in Amazon S3 utilizzando SQL standard. Questo modello utilizza Athena per interrogare e trasformare i dati JSON. Con poche azioni in Console di gestione AWS, puoi indirizzare Athena ai tuoi dati in Amazon S3 e utilizzare SQL standard per eseguire query singole. Athena è serverless, quindi non c'è alcuna infrastruttura da configurare o gestire e paghi solo per le query che esegui. Athena si ridimensiona automaticamente ed esegue le query in parallelo, quindi i risultati sono rapidi, anche con set di dati di grandi dimensioni e query complesse.
- Amazon Quick Sight è un servizio di business intelligence (BI) su scala cloud che ti aiuta a visualizzare, analizzare e generare report sui dati in un'unica dashboard. Quick Sight ti consente di creare e pubblicare facilmente dashboard interattive che includono approfondimenti sull'apprendimento automatico (ML). Puoi accedere a queste dashboard da qualsiasi dispositivo e incorporarle nelle tue applicazioni, portali e siti Web.

Esempio di codice

Il seguente file JSON fornisce una struttura di dati annidata che è possibile utilizzare in questo modello.

```
{
  "symbol": "AAPL",
  "financials": [
    {
      "reportDate": "2017-03-31",
```

```
      "grossProfit": 20591000000,
      "costOfRevenue": 32305000000,
      "operatingRevenue": 52896000000,
      "totalRevenue": 52896000000,
      "operatingIncome": 14097000000,
      "netIncome": 11029000000,
      "researchAndDevelopment": 2776000000,
      "operatingExpense": 6494000000,
      "currentAssets": 101990000000,
      "totalAssets": 334532000000,
      "totalLiabilities": 200450000000,
      "currentCash": 15157000000,
      "currentDebt": 13991000000,
      "totalCash": 67101000000,
      "totalDebt": 98522000000,
      "shareholderEquity": 134082000000,
      "cashChange": -1214000000,
      "cashFlow": 12523000000,
      "operatingGainsLosses": null
    }
  ]
}
```

Epiche

Configura un bucket S3

Operazione	Description	Competenze richieste
Crea un bucket S3.	Per creare un bucket per archiviare il file JSON, accedi a Console di gestione AWS, apri la console Amazon S3, quindi scegli Crea bucket. Per ulteriori informazioni, consulta Creazione di un bucket nella documentazione di Amazon S3.	Amministratore di sistema
Aggiungi i dati JSON annidati.	Carica il tuo file JSON nel bucket S3. Per un file JSON di	Amministratore di sistema

Operazione	Description	Competenze richieste
	esempio, consulta la sezione precedente. Per istruzioni, consulta Caricamento di oggetti nella documentazione di Amazon S3.	

Analizza i dati in Athena

Operazione	Description	Competenze richieste
Crea una tabella per mappare i dati JSON.	1. Apri la console Athena. 2. Crea un database seguendo le istruzioni contenute nella documentazione di Athena. 3. Dal menu Database, scegli il database che hai creato. 4. Nell'editor di query, inserisci un'`CREATE TABLE`istruzione come la seguente: `CREATE EXTERNAL TABLE financials_json ( symbol string, financials array< struct<reportdate: string, grossprofit: bigint, totalrevenue: bigint, totalcash: bigint, totaldebt: bigint,`	Developer

Operazione	Description	Competenze richieste

	<pre>researcha
	nddevelopment:
	 bigint>>
	)
	ROW FORMAT SERDE
	 'org.openx.data.js
	onserde.JsonSerDe'
	 LOCATION 's3://s3b
	ucket-for-athena/'</pre> dove `LOCATION` specifica la posizione del bucket S3 che contiene il file JSON.
	5. Scegli Esegui per creare la tabella.
	
	Per ulteriori informazioni sulla creazione di tabelle, consulta la documentazione di Athena.	

Operazione	Description	Competenze richieste
Crea una vista per l'analisi dei dati.	1. Apri la console Athena. 2. Crea un database seguendo le istruzioni contenute nella documenta zione di Athena. 3. Dal menu Database, scegli il database che hai creato. 4. Nell'editor di query, inserisci un'`CREATE VIEW`istruzione come la seguente: 5. Scegli Run (Esegui) per creare la visualizzazione.	Developer

```
CREATE OR REPLACE VIEW
 financial_json_view
 AS
  SELECT
  symbol,
  financials[1].repo
rtdate one_repor
t_date, -- indexes
 start with 1
  financials[1].tota
lrevenue one_total
_revenue,
  financials[1].repo
rtdate another_r
eport_date,
  financials[1].tota
lrevenue another_t
otal_revenue
FROM financials_json
 where symbol='AAPL'
ORDER BY
  1
```

Operazione	Description	Competenze richieste
	Per ulteriori informazioni sulla creazione di viste, consulta la documentazione di Athena.	
Analizza e convalida i dati.	1. Apri la console Athena. 2. Nell'editor delle query, esegui le interrogazioni utilizzando la vista creata nel passaggio precedente. 3. Convalida i dati rispetto al file JSON, per confermare che i nomi delle colonne e i tipi di dati siano mappati correttamente.	Developer

Visualizza i dati in Quick Sight

Operazione	Description	Competenze richieste
Configura Athena come fonte di dati in Quick Sight.	1. Apri la console Quick. 2. Scegli Set di dati, Nuovo set di dati. 3. Scegli Athena come fonte di dati. 4. Scegli il database che include la vista che hai creato. 5. Scegli la vista per cui vuoi creare un set di dati. 6. Nella pagina Termina la creazione del set di dati, scegli Interroga direttamente i tuoi dati.	Amministratore di sistema

Operazione	Description	Competenze richieste
	7. Scegliere Visualize (Visualiz za).	
Visualizza i dati in Quick Sight.	1. Dopo aver visualizzato il set di dati, scegli gli elementi visivi dal riquadro di sinistra e scegli i campi per il set di dati. Per ulteriori informazi oni, consulta il tutorial nella documentazione rapida. 2. Salvate le modifiche all'analisi. 3. Scegli Pubblica dashboard per pubblicare gli elementi visivi che hai creato.	Analista dei dati

Risorse correlate

- Documentazione Amazon Athena
- Tutorial Amazon Quick Sight
- Lavorare con JSON annidato (post sul blog)

Automatizza l'inserimento dei dati da Amazon S3 AWS Data Exchange

Adnan Alvee e Manikanta Gona, Amazon Web Services

Riepilogo

Questo modello fornisce un CloudFormation modello che ti consente di inserire automaticamente i dati dal tuo data lake AWS Data Exchange in Amazon Simple Storage Service (Amazon S3).

AWS Data Exchange è un servizio che semplifica lo scambio sicuro di set di dati basati su file nel cloud AWS. AWS Data Exchange i set di dati sono basati su abbonamento. In qualità di abbonato, puoi anche accedere alle revisioni dei set di dati man mano che i provider pubblicano nuovi dati.

Il CloudFormation modello crea un evento in Amazon CloudWatch Events e una AWS Lambda funzione. L'evento rileva eventuali aggiornamenti al set di dati a cui ti sei abbonato. Se è presente un aggiornamento, CloudWatch avvia una funzione Lambda, che copia i dati nel bucket S3 specificato. Quando i dati sono stati copiati correttamente, Lambda ti invia una notifica Amazon Simple Notification Service (Amazon SNS).

Prerequisiti e limitazioni

Prerequisiti

- Una attiva Account AWS
- Sottoscrizione a un set di dati in AWS Data Exchange

Limitazioni

- Il CloudFormation modello deve essere distribuito separatamente per ogni set di dati sottoscritto in. AWS Data Exchange

Architecture

Stack tecnologico Target

- AWS Lambda
- Simple Storage Service (Amazon S3)

- AWS Data Exchange
- Amazon CloudWatch
- Amazon SNS

Architettura Target

Automazione e scalabilità

Puoi utilizzare il CloudFormation modello più volte per i set di dati che desideri inserire nel data lake.

Tools (Strumenti)

- AWS Data Exchangeconsente AWS ai clienti di scambiare facilmente in modo sicuro set di dati basati su file in. Cloud AWS In qualità di abbonato, puoi trovare e abbonarti a centinaia di prodotti di fornitori di dati qualificati. Quindi, puoi scaricare rapidamente il set di dati o copiarlo su Amazon S3 per utilizzarlo in una varietà di servizi di AWS analisi e apprendimento automatico. Chiunque abbia un Account AWS può essere AWS Data Exchange abbonato.
- AWS Lambda consente di eseguire il codice senza effettuare il provisioning dei server o senza gestirli. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo. Paghi solo per il tempo di elaborazione che consumi; non ci sono costi quando il codice non è in esecuzione. Con Lambda, puoi eseguire codice praticamente per qualsiasi tipo di applicazione o servizio di backend senza alcuna amministrazione. Lambda esegue il codice su un'infrastruttura di elaborazione ad alta disponibilità e gestisce tutte le risorse di calcolo, tra cui la manutenzione di server e sistemi operativi, il provisioning della capacità e il ridimensionamento automatico, il monitoraggio del codice e la registrazione.
- Amazon S3 fornisce storage per Internet. È possibile utilizzare Amazon S3 per memorizzare e recuperare qualsiasi volume di dati, in qualunque momento e da qualunque luogo tramite il Web.
- Amazon CloudWatch Events offre un flusso quasi in tempo reale di eventi di sistema che descrivono i cambiamenti nelle AWS risorse. Utilizzando semplici regole che puoi configurare rapidamente, puoi abbinare gli eventi e indirizzarli a una o più funzioni o flussi di destinazione. CloudWatch Gli eventi vengono a conoscenza dei cambiamenti operativi man mano che si verificano. Risponde a questi cambiamenti operativi e adotta le azioni correttive necessarie, inviando messaggi per rispondere all'ambiente, attivando funzioni, apportando modifiche e acquisendo informazioni sullo stato. Puoi anche utilizzare CloudWatch Events per pianificare azioni

automatiche che si avviano automaticamente in determinati momenti utilizzando le espressioni cron o rate.

- Amazon Simple Notification Service (Amazon SNS) consente alle applicazioni, agli utenti finali e ai dispositivi di inviare e ricevere istantaneamente notifiche dal cloud. Amazon SNS fornisce argomenti (canali di comunicazione) per la messaggistica ad alto throughput e basata su push. many-to-many Utilizzando gli argomenti di Amazon SNS, gli editori possono distribuire messaggi a un gran numero di abbonati per l'elaborazione parallela, tra cui code Amazon Simple Queue Service (Amazon SQS), funzioni Lambda e webhook. HTTP/S Puoi anche utilizzare Amazon SNS per inviare notifiche agli utenti finali tramite push, SMS ed e-mail mobili.

Epiche

Iscriviti a un set di dati

Operazione	Description	Competenze richieste
Abbonarsi a un set di dati.	Nella AWS Data Exchange console, sottoscrivi un set di dati. Per istruzioni, consulta Sottoscrizione ai prodotti di dati AWS Data Exchange nella AWS documentazione.	Informazioni generali su AWS
Nota gli attributi del set di dati.	Annota Regione AWS l'ID e l'ID di revisione del set di dati. Ti servirà per il CloudForm ation modello nel passaggio successivo.	Informazioni generali su AWS

Implementa il modello CloudFormation

Operazione	Description	Competenze richieste
Crea un bucket e una cartella S3.	Se disponi già di un data lake in Amazon S3, crea una cartella in cui archiviare i dati	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	da cui importare. AWS Data Exchange Se stai distribuendo il modello a scopo di test, crea un nuovo bucket S3 e annota il nome del bucket e il prefisso della cartella per il passaggio successivo.	

Operazione	Description	Competenze richieste
Implementa il CloudFormation modello.	Distribuisci il CloudForm ation modello fornito come allegato a questo modello. Per istruzioni, consulta la CloudFormation documenta zione. Configura i seguenti parametri in modo che corrispondano alle impostazioni Account AWS del set di dati e del bucket S3: Dataset Regione AWS, Dataset ID, Revision ID, S3 Bucket Name (ad esempio**`DOC-EXAMPLE-BUCKET`**), Folder Prefix (ad esempio) ed Email for SNS Notification. **`myfolder/`** È possibile impostare il parametro Dataset Name su qualsiasi nome. Quando si distribuisce il modello, esegue una funzione Lambda per importare automaticamente il primo set di dati disponibi le nel set di dati. L'ingestione successiva avviene quindi automaticamente, non appena arrivano nuovi dati nel set di dati.	Informazioni generali su AWS

Risorse correlate

- Abbonamento ai prodotti di dati su AWS Data Exchange (AWS Data Exchange documentazione)

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Automatizza l'applicazione della crittografia in AWS Glue utilizzando un modello AWS CloudFormation

Diogo Guedes, Amazon Web Services

Riepilogo

Questo modello mostra come configurare e automatizzare l'applicazione della crittografia in AWS Glue utilizzando un CloudFormation modello AWS. Il modello crea tutte le configurazioni e le risorse necessarie per applicare la crittografia. Queste risorse includono una configurazione iniziale, un controllo preventivo creato da una EventBridge regola Amazon e una funzione AWS Lambda.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Autorizzazioni per distribuire il CloudFormation modello e le relative risorse

Limitazioni

Questo controllo di sicurezza è regionale. Devi implementare il controllo di sicurezza in ogni regione AWS in cui desideri configurare l'applicazione della crittografia in AWS Glue.

Architecture

Stack tecnologico Target

- Amazon CloudWatch Logs (da AWS Lambda)
- EventBridge Regola Amazon
- CloudFormation Stack AWS
- AWS CloudTrail
- Ruolo e policy gestiti da AWS Identity and Access Management (IAM)
- AWS Key Management Service (AWS KMS)
- Alias AWS KMS
- Funzione AWS Lambda

- AWS Systems Manager Parameter Store

Architettura Target

Il diagramma seguente mostra come automatizzare l'applicazione della crittografia in AWS Glue.

Il diagramma mostra il flusso di lavoro seguente:

1. Un CloudFormation modello crea tutte le risorse, inclusa la configurazione iniziale e il controllo investigativo per l'applicazione della crittografia in AWS Glue.
2. Una EventBridge regola rileva un cambiamento di stato nella configurazione di crittografia.
3. Viene richiamata una funzione Lambda per la valutazione e la registrazione tramite Logs. CloudWatch In caso di rilevamento non conforme, il Parameter Store viene ripristinato con un Amazon Resource Name (ARN) per una chiave AWS KMS. Il servizio viene ripristinato allo stato conforme con la crittografia abilitata.

Automazione e scalabilità

Se utilizzi AWS Organizations, puoi utilizzare AWS CloudFormation StackSets per distribuire questo modello in più account in cui desideri abilitare l'applicazione della crittografia in AWS Glue.

Tools (Strumenti)

- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue risorse AWS e delle applicazioni che esegui su AWS in tempo reale.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, funzioni Lambda, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altri account AWS.
- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account e regioni AWS.
- AWS ti CloudTrail aiuta a abilitare il controllo operativo e dei rischi, la governance e la conformità del tuo account AWS.
- AWS Glue è un servizio di estrazione, trasformazione e caricamento (ETL) completamente gestito. Ti aiuta a classificare, pulire, arricchire e spostare i dati in modo affidabile tra archivi di dati e flussi di dati.

- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- AWS Systems Manager ti aiuta a gestire le applicazioni e l'infrastruttura in esecuzione nel cloud AWS. Semplifica la gestione delle applicazioni e delle risorse, riduce i tempi di rilevamento e risoluzione dei problemi operativi e ti aiuta a gestire le tue risorse AWS in modo sicuro su larga scala.

Codice

Il codice per questo pattern è disponibile nel repository -driven. GitHub aws-custom-guardrail-event

Best practice

AWS Glue supporta la crittografia dei dati a riposo per la creazione di lavori in AWS Glue e lo sviluppo di script utilizzando endpoint di sviluppo.

Prendi in considerazione le seguenti best practice:

- Configura i job ETL e gli endpoint di sviluppo per utilizzare le chiavi AWS KMS per scrivere dati crittografati inattivi.
- Crittografa i metadati archiviati nel catalogo dati di AWS Glue utilizzando chiavi gestite tramite AWS KMS.
- Usa le chiavi AWS KMS per crittografare i segnalibri dei lavori e i log generati dai crawler e dai job ETL.

Epiche

Avvia il modello CloudFormation

Operazione	Description	Competenze richieste
Implementa il CloudFormation modello.	Scarica il `aws-custom-guardrail-event-`	Architetto del cloud

Operazione	Description	Competenze richieste
	`driven.yaml` modello dal GitHub repository, quindi distribuiscilo. Lo `CREATE_COMPLETE` stato indica che il modello è stato distribuito correttamente. **Note** Il modello non richiede parametri di input.	

Verifica le impostazioni di crittografia in AWS Glue

Operazione	Description	Competenze richieste
Controlla le configurazioni delle chiavi AWS KMS.	1. Accedi alla Console di gestione AWS e quindi apri la console AWS Glue. 2. Nel pannello di navigazione, in Data Catalog, scegli Impostazioni del catalogo. 3. Verifica che le impostazioni per la crittografia dei metadati e la crittografia delle password di connessione siano contrassegnate e configurate per l'uso. `KMSKeyGlue`	Architetto del cloud

Verifica l'applicazione della crittografia

Operazione	Description	Competenze richieste
Identifica l'impostazione di crittografia in CloudFormation.	1. Accedi alla Console di gestione AWS e apri la CloudFormation console. 2. Nel pannello di navigazione, scegli Stacks, quindi scegli il tuo stack. 3. Scegli la scheda Risorse. 4. Nella tabella Risorse, trova l'impostazione di crittografia per Logical ID.	Architetto del cloud
Passa l'infrastruttura predisposta a uno stato non conforme.	1. Accedi alla Console di gestione AWS e quindi apri la console AWS Glue. 2. Nel pannello di navigazione, in Data Catalog, scegli Impostazioni del catalogo. 3. Deseleziona la casella di controllo Crittografia dei metadati. 4. Deselezionate la casella di controllo Crittografa le password di connessione. 5. Scegli Save (Salva). 6. Aggiorna la console AWS Glue. Il guardrail rileva lo stato di non conformità in AWS Glue dopo aver deselezionato le caselle di controllo,	Architetto del cloud

Operazione	Description	Competenze richieste
	quindi impone la conformità correggendo automaticamente l'errata configurazione della crittografia. Di conseguenza, le caselle di controllo per la crittografia devono essere nuovamente selezionate dopo aver aggiornato la pagina.	

Risorse correlate

- Creazione di uno stack sulla CloudFormation console AWS (CloudFormation documentazione AWS)
- Creazione di una regola CloudWatch Events che si attiva su una chiamata API AWS utilizzando AWS CloudTrail (CloudWatch documentazione Amazon)
- Configurazione della crittografia in AWS Glue (documentazione AWS Glue)

Crea una pipeline di dati per importare, trasformare e analizzare i dati di Google Analytics utilizzando il AWS DataOps Development Kit

Anton Kukushkin e Rudy Puig, Amazon Web Services

Riepilogo

Questo modello descrive come creare una pipeline di dati per importare, trasformare e analizzare i dati di Google Analytics utilizzando il AWS DataOps Development Kit (DDK) e altro.AWS Servizi AWS Il AWS DDK è un framework di sviluppo open source che ti aiuta a creare flussi di lavoro di dati e un'architettura di dati moderna su. AWS Uno degli obiettivi principali del AWS DDK è quello di farti risparmiare tempo e fatica tipicamente dedicati ad attività di pipeline di dati ad alta intensità di manodopera, come l'orchestrazione delle pipeline, la creazione di infrastrutture e la creazione dell'infrastruttura alla base di tale infrastruttura. DevOps Puoi affidare queste attività ad alta intensità di lavoro a AWS DDK in modo da poterti concentrare sulla scrittura di codice e su altre attività di alto valore.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un AppFlow connettore Amazon per Google Analytics, configurato
- Python e pip (gestore di pacchetti di Python)
- Git, installato e configurato
- AWS Command Line Interface (AWS CLI), installato e configurato
- AWS Cloud Development Kit (AWS CDK), installato

Versioni del prodotto

- Python 3.7 o versioni successive
- pip 9.0.3 o versioni successive

Architecture

Stack tecnologico

- Amazon AppFlow
- Amazon Athena
- Amazon CloudWatch
- Amazon EventBridge
- Amazon Simple Storage Service (Amazon S3)
- Amazon Simple Queue Service (Amazon SQS)
- AWS DataOps Kit di sviluppo (AWS DDK)
- AWS Lambda

Architettura Target

Il diagramma seguente mostra il processo basato sugli eventi che acquisisce, trasforma e analizza i dati di Google Analytics.

Il diagramma mostra il flusso di lavoro seguente:

1. Una regola per gli eventi CloudWatch pianificati di Amazon richiama Amazon. AppFlow
2. Amazon AppFlow inserisce i dati di Google Analytics in un bucket S3.
3. Dopo che i dati sono stati inseriti dal bucket S3, EventBridge vengono generate notifiche di eventi, acquisite da una regola CloudWatch Events e quindi inserite in una coda Amazon SQS.
4. Una funzione Lambda consuma gli eventi dalla coda Amazon SQS, legge i rispettivi oggetti S3, trasforma gli oggetti in formato Apache Parquet, scrive gli oggetti trasformati nel bucket S3 e quindi crea o aggiorna la definizione della tabella. AWS Glue Data Catalog
5. Una query Athena viene eseguita sulla tabella.

Tools (Strumenti)

AWS strumenti

- Amazon AppFlow è un servizio di integrazione completamente gestito che consente di scambiare dati in modo sicuro tra applicazioni SaaS (Software as a Service).

- Amazon Athena è un servizio di query interattivo che ti aiuta ad analizzare i dati direttamente in Amazon S3 utilizzando SQL standard.
- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue AWS risorse e delle applicazioni su cui esegui AWS in tempo reale.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altro modo. Account AWS
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Simple Queue Service (Amazon SQS) fornisce una coda ospitata sicura, durevole e disponibile che ti aiuta a integrare e disaccoppiare sistemi e componenti software distribuiti.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS Cloud Development Kit (AWS CDK)è un framework per definire l'infrastruttura cloud in codice e fornirla tramite. CloudFormation
- AWS DataOps Development Kit (AWS DDK) è un framework di sviluppo open source che ti aiuta a creare flussi di lavoro di dati e un'architettura di dati moderna su. AWS

Codice

Il codice per questo pattern è disponibile negli archivi GitHub AWS DataOps Development Kit (AWS DDK) e Analyzing Google Analytics with Amazon AppFlow, Amazon Athena AWS DataOps e Development Kit.

Epiche

Prepara l'ambiente

Operazione	Description	Competenze richieste
Clona il codice sorgente.	Per clonare il codice sorgente, esegui il seguente comando:	DevOps ingegnere

Operazione	Description	Competenze richieste
	`git clone https://github.com/aws-samples/aws-ddk-examples.git`	
Crea un ambiente virtuale.	Passa alla directory del codice sorgente, quindi esegui il comando seguente per creare un ambiente virtuale: `cd google-analytics-data-using-appflow/python && python3 -m venv .venv`	DevOps ingegnere
Installa le dipendenze.	Per attivare l'ambiente virtuale e installare le dipendenze, esegui il seguente comando: `source .venv/bin/activate && pip install -r requirements.txt`	DevOps ingegnere

Implementa l'applicazione che utilizza la tua pipeline di dati

Operazione	Description	Competenze richieste
Avvia l'ambiente.	1. Conferma che AWS CLI sia configurato con credenziali valide per il tuo Account AWS Per ulteriori informazioni, consulta la sezione Utilizzo di profili denominati nella AWS CLI documentazione.	DevOps ingegnere

Operazione	Description	Competenze richieste
	2. Esegui il comando `cdk bootstrap --profile [AWS_PROFILE] .`	
Distribuisci i dati.	Per distribuire la pipeline di dati, esegui il comando. `cdk deploy --profile [AWS_PROFILE]`	DevOps ingegnere

Test della distribuzione

Operazione	Description	Competenze richieste
Convalida lo stato dello stack.	1. Apri la CloudFormation console. 2. Nella pagina Stacks, verifica che lo stato dello stack sia. `DdkAppflowAthenaStack CREATE_COMPLETE`	DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
La distribuzione non riesce durante la creazione di una `AWS::AppFlow::Flow` risorsa e viene visualizzato il seguente errore: `Connector Profile with name ga-connection does not exist`	Conferma di aver creato un AppFlow connettore Amazon per Google Analytics e di avergli dato un nome `ga-connection` . Per istruzioni, consulta Google Analytics nella AppFlow documentazione di Amazon.

Risorse correlate

- AWS DataOps Kit di sviluppo (AWS DDK) (GitHub)
- AWS Esempi DDK () GitHub

Informazioni aggiuntive

AWS Le pipeline di dati DDK sono composte da una o più fasi. Nei seguenti esempi di codice, li usi `AppFlowIngestionStage` per importare dati da Google Analytics, `SqsToLambdaStage` gestire la trasformazione dei dati ed `AthenaSQLStage` eseguire la query Athena.

Innanzitutto, vengono create le fasi di trasformazione e ingestione dei dati, come mostra il seguente esempio di codice:

```
        appflow_stage = AppFlowIngestionStage(
            self,
            id="appflow-stage",
            flow_name=flow.flow_name,
        )
        sqs_lambda_stage = SqsToLambdaStage(
            self,
            id="lambda-stage",
            lambda_function_props={
                "code": Code.from_asset("./ddk_app/lambda_handlers"),
                "handler": "handler.lambda_handler",
                "layers": [
                    LayerVersion.from_layer_version_arn(
                        self,
                        id="layer",
                        layer_version_arn=f"arn:aws:lambda:
{self.region}:336392948345:layer:AWSDataWrangler-Python39:1",
                    )
                ],
                "runtime": Runtime.PYTHON_3_9,
            },
        )
        # Grant lambda function S3 read & write permissions
        bucket.grant_read_write(sqs_lambda_stage.function)
        # Grant Glue database & table permissions
        sqs_lambda_stage.function.add_to_role_policy(
            self._get_glue_db_iam_policy(database_name=database.database_name)
```

```
        )
        athena_stage = AthenaSQLStage(
            self,
            id="athena-sql",
            query_string=[
                (
                    "SELECT year, month, day, device, count(user_count) as cnt "
                    f"FROM {database.database_name}.ga_sample "
                    "GROUP BY year, month, day, device "
                    "ORDER BY cnt DESC "
                    "LIMIT 10; "
                )
            ],
            output_location=Location(
                bucket_name=bucket.bucket_name, object_key="query-results/"
            ),
            additional_role_policy_statements=[
                self._get_glue_db_iam_policy(database_name=database.database_name)
            ],
        )
```

Successivamente, il `DataPipeline` costrutto viene utilizzato per «collegare» gli stadi utilizzando EventBridge delle regole, come mostra il seguente esempio di codice:

```
        (
            DataPipeline(self, id="ingestion-pipeline")
            .add_stage(
                stage=appflow_stage,
                override_rule=Rule(
                    self,
                    "schedule-rule",
                    schedule=Schedule.rate(Duration.hours(1)),
                    targets=appflow_stage.targets,
                ),
            )
            .add_stage(
                stage=sqs_lambda_stage,
                # By default, AppFlowIngestionStage stage emits an event after the flow
run finishes successfully
                # Override rule below changes that behavior to call the the stage when
data lands in the bucket instead
                override_rule=Rule(
                    self,
```

```
                "s3-object-created-rule",
                event_pattern=EventPattern(
                    source=["aws.s3"],
                    detail={
                        "bucket": {"name": [bucket.bucket_name]},
                        "object": {"key": [{"prefix": "ga-data"}]},
                    },
                    detail_type=["Object Created"],
                ),
                targets=sqs_lambda_stage.targets,
            ),
        )
        .add_stage(stage=athena_stage)
    )
```

Per altri esempi di codice, consulta il GitHub repository Analisi dei dati di Google Analytics con Amazon AppFlow, Amazon Athena AWS DataOps e Development Kit.

Crea una pipeline di elaborazione video utilizzando Amazon Kinesis Video Streams e AWS Fargate

Piotr Chotkowski e Pushparaju Thangavel, Amazon Web Services

Riepilogo

Questo modello dimostra come utilizzare Amazon Kinesis Video Streams e AWS Fargate per estrarre fotogrammi da un flusso video e archiviarli come file di immagine per un'ulteriore elaborazione in Amazon Simple Storage Service (Amazon S3).

Il pattern fornisce un'applicazione di esempio sotto forma di progetto Java Maven. Questa applicazione definisce l'infrastruttura AWS utilizzando l'AWS Cloud Development Kit (AWS CDK). Sia la logica di elaborazione dei frame che le definizioni dell'infrastruttura sono scritte nel linguaggio di programmazione Java. È possibile utilizzare questa applicazione di esempio come base per sviluppare la propria pipeline di elaborazione video in tempo reale o per creare la fase di preelaborazione video di una pipeline di apprendimento automatico.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Java SE Development Kit (JDK) 11, installato
- Apache Maven, installato
- AWS Cloud Development Kit (AWS CDK), installato
- AWS Command Line Interface (AWS CLI) versione 2, installata
- Docker (necessario per creare immagini Docker da utilizzare nelle definizioni delle attività di AWS Fargate), installato

Limitazioni

Questo modello è inteso come dimostrazione di concetto o come base per un ulteriore sviluppo. Non dovrebbe essere utilizzato nella sua forma attuale nelle installazioni di produzione.

Versioni del prodotto

- Questo modello è stato testato con la versione CDK AWS 1.77.0 (vedi versioni AWS CDK)

- JDK 11
- AWS CLI versione 2

Architecture

Stack tecnologico Target

- Amazon Kinesis Video Streams
- Attività AWS Fargate
- Coda Amazon Simple Queue Service (Amazon SQS)
- Bucket Amazon S3

Architettura Target

L'utente crea un flusso video Kinesis, carica un video e invia un messaggio JSON contenente dettagli sul flusso video Kinesis in ingresso e sul bucket S3 di uscita in una coda SQS. AWS Fargate, che esegue l'applicazione principale in un contenitore, estrae il messaggio dalla coda SQS e inizia a estrarre i frame. Ogni frame viene salvato in un file di immagine e archiviato nel bucket S3 di destinazione.

Automazione e scalabilità

L'applicazione di esempio può essere scalata sia orizzontalmente che verticalmente all'interno di una singola regione AWS. La scalabilità orizzontale può essere ottenuta aumentando il numero di attività AWS Fargate distribuite che leggono dalla coda SQS. La scalabilità verticale può essere ottenuta aumentando il numero di thread di suddivisione dei frame e di pubblicazione di immagini nell'applicazione. Queste impostazioni vengono passate come variabili di ambiente all'applicazione nella definizione della QueueProcessingFargateServicerisorsa nell'AWS CDK. A causa della natura della distribuzione dello stack AWS CDK, puoi distribuire questa applicazione in più regioni e account AWS senza sforzi aggiuntivi.

Tools (Strumenti)

Strumenti

- AWS CDK è un framework di sviluppo software per definire l'infrastruttura e le risorse cloud utilizzando linguaggi di programmazione come Python TypeScript JavaScript, Java e C#/.Net.

- Amazon Kinesis Video Streams è un servizio AWS completamente gestito che puoi utilizzare per lo streaming di video in diretta dai dispositivi al cloud AWS o creare applicazioni per l'elaborazione video in tempo reale o l'analisi video orientata ai batch.
- AWS Fargate è un motore di elaborazione serverless per container. Fargate elimina la necessità di effettuare il provisioning e la gestione dei server e consente di concentrarsi sullo sviluppo delle applicazioni.
- Amazon S3 è un servizio di storage di oggetti che offre scalabilità, disponibilità dei dati, sicurezza e prestazioni.
- Amazon SQS è un servizio di accodamento dei messaggi completamente gestito che consente di disaccoppiare e scalare microservizi, sistemi distribuiti e applicazioni serverless.

Codice

- È allegato un file.zip del progetto applicativo di esempio (). `frame-splitter-code.zip`

Epiche

Implementa l'infrastruttura

Operazione	Description	Competenze richieste
Avvia il daemon Docker.	Avvia il demone Docker sul tuo sistema locale. L'AWS CDK utilizza Docker per creare l'immagine utilizzata nel task AWS Fargate. È necessario eseguire Docker prima di procedere al passaggio successivo.	Sviluppatore, DevOps ingegnere
Compilare il progetto.	Scaricate l'applicazione di `frame-splitter-code` esempio (allegata) ed estraetene il contenuto in una cartella sul computer locale. Prima di poter implementare	Sviluppatore, DevOps ingegnere

Operazione	Description	Competenze richieste
	l'infrastruttura, è necessario creare il progetto Java Maven. Al prompt dei comandi, accedete alla directory principale del progetto e create il progetto eseguendo il comando: `mvn clean install`	
Avvia il CDK AWS.	(Solo utenti AWS CDK per la prima volta) Se è la prima volta che utilizzi il CDK AWS, potresti dover avviare l'ambiente eseguendo il comando AWS CLI: `cdk bootstrap --profile "$AWS_PROFILE_NAME"` where `$AWS_PROFILE_NAME` contiene il nome del profilo AWS contenuto nelle tue credenziali AWS. In alternativa, puoi rimuovere questo parametro per utilizzare il profilo predefinito. Per ulteriori informazioni, consulta la documentazione di AWS CDK.	Sviluppatore, DevOps ingegnere

Operazione	Description	Competenze richieste
Implementa lo stack CDK AWS.	In questa fase, crei le risorse di infrastruttura richieste (coda SQS, bucket S3, definizione di attività AWS Fargate) nel tuo account AWS, crei l'immagine Docker necessaria per il task AWS Fargate e distribuisci l'applicazione. Al prompt dei comandi, accedi alla directory principale del progetto ed esegui il comando: `cdk deploy --profile "$AWS_PROFILE_NAME" --all` where `$AWS_PROFILE_NAME` contiene il nome del profilo AWS contenuto nelle tue credenziali AWS. In alternativa, puoi rimuovere questo parametro per utilizzare il profilo predefinito. Conferma la distribuzione. Annota i valori QueueUrle Bucket dall'output della distribuzione CDK; ti serviranno nei passaggi successivi. L'AWS CDK crea gli asset, li carica sul tuo account AWS e crea tutte le risorse dell'infrastruttura. Puoi osservare il processo di creazione delle risorse nella CloudFormation console AWS. Per ulteriori	Sviluppatore, DevOps ingegnere

Operazione	Description	Competenze richieste
	informazioni, consulta la documentazione di AWS e la CloudFormation documentazione di AWS CDK.	

Operazione	Description	Competenze richieste
Crea un flusso video.	In questo passaggio, crei un flusso video Kinesis che fungerà da flusso di input per l'elaborazione video. Assicurati di avere la CLI AWS installata e configurata. Nella CLI di AWS, esegui: `aws kinesisvideo --profile "$AWS_PROFILE_NAME" create-stream --stream-name "$STREAM_NAME" --data-retention-in-hours "24"` where `$AWS_PROFILE_NAME` contiene il nome del profilo AWS dalle tue credenziali AWS (o rimuovi questo parametro per utilizzare il profilo predefinito) ed `$STREAM_NAME` è qualsiasi nome di stream valido. In alternativa, puoi creare un flusso video utilizzando la console Kinesis seguendo i passaggi nella documentazione di Kinesis Video Streams. Prendi nota dell'AWS Resource Name (ARN) dello stream creato; ti servirà in seguito.	Sviluppatore, DevOps ingegnere

Esegui un esempio

Operazione	Description	Competenze richieste
Carica il video nello stream.	Nella cartella del progetto dell'`frame-splitter-code` applicazione di esempio, aprite il `ProcessingTaskTest.java` file contenuto nella `src/test/java/amazon/awscdk/examples/splitter` cartella. Sostituite le `streamName` variabili `profileName` and con i valori utilizzati nei passaggi precedenti. Per caricare il video di esempio nello stream video Kinesis creato nel passaggio precedente, esegui: `amazon.awscdk.examples.splitter.ProcessingTaskTest#testExample test` In alternativa, puoi caricare il tuo video utilizzando uno dei metodi descritti nella documentazione di Kinesis Video Streams.	Sviluppatore, ingegnere DevOps
Avvia l'elaborazione video.	Ora che hai caricato un video nello stream video di Kinesis, puoi iniziare a elaborarlo. Per avviare la logica di	Sviluppatore, ingegnere DevOps

Operazione	Description	Competenze richieste

	elaborazione, devi inviare un messaggio con i dettagli alla coda SQS creata dal CDK AWS durante la distribuzione. Per inviare un messaggio utilizzando la CLI di AWS, esegui:
	
	`aws sqs --profile "$AWS_PROFILE_NAME" send-message --queue-url QUEUE_URL --message-body MESSAGE`
	
	where `$AWS_PROFILE_NAME` contiene il nome del profilo AWS dalle tue credenziali AWS (rimuovi questo parametro per utilizzare il profilo predefinito), `QUEUE_URL` è il QueueUrlvalore dell'output di AWS CDK ed `MESSAGE` è una stringa JSON nel seguente formato:
	
	`{ "streamARN": "STREAM_ARN", "bucket": "BUCKET_NAME", "s3Directory": "test-output" }`
	
	dove `STREAM_ARN` è l'ARN del flusso video creato in un passaggio precedente ed `BUCKET_NAME` è il valore	

Operazione	Description	Competenze richieste
	Bucket dall'output di AWS CDK. L'invio di questo messaggio avvia l'elaborazione video. In alternativa, puoi inviare un messaggio utilizzando la console Amazon SQS, come descritto nella documenta zione di Amazon SQS.	
Visualizza le immagini dei fotogrammi video.	Puoi vedere le immagini risultanti nel bucket di output S3 `s3://BUCKET_NAME/test-output` dove si `BUCKET_NAME` trova il valore del bucket dall'output di AWS CDK.	Sviluppatore, ingegnere DevOps

Risorse correlate

- Documentazione CDK AWS
- Riferimento all'API CDK AWS
- Workshop introduttivo su AWS CDK
- Documentazione di Amazon Kinesis Video Streams
- Esempio: identificazione di oggetti nei flussi video mediante SageMaker
- Esempio: analisi e rendering dei frammenti di Kinesis Video Streams
- Analizza video live su larga scala in tempo reale utilizzando Amazon Kinesis Video Streams SageMaker e Amazon (post sul blog di AWS Machine Learning)
- Nozioni di base su AWS Fargate

Informazioni aggiuntive

Scelta di un IDE

Ti consigliamo di utilizzare il tuo IDE Java preferito per creare ed esplorare questo progetto.

Pulizia

Al termine dell'esecuzione di questo esempio, rimuovi tutte le risorse distribuite per evitare costi aggiuntivi dell'infrastruttura AWS.

Per rimuovere l'infrastruttura e il flusso video, usa questi due comandi nella CLI di AWS:

```
cdk destroy --profile "$AWS_PROFILE_NAME" --all
```

```
aws kinesisvideo --profile "$AWS_PROFILE_NAME" delete-stream --stream-arn "$STREAM_ARN"
```

In alternativa, puoi rimuovere le risorse manualmente utilizzando la CloudFormation console AWS per rimuovere lo CloudFormation stack AWS e la console Kinesis per rimuovere il flusso video Kinesis. Tieni presente che `cdk destroy` non rimuove il bucket S3 di output o le immagini nei repository Amazon Elastic Container Registry (Amazon ECR) (). `aws-cdk/assets` È necessario rimuoverli manualmente.

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Crea una pipeline di servizi ETL per caricare i dati in modo incrementale da Amazon S3 ad Amazon Redshift utilizzando AWS Glue

Rohan Jamadagni e Arunabha Datta, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni su come configurare Amazon Simple Storage Service (Amazon S3) per prestazioni ottimali del data lake e quindi caricare modifiche incrementali ai dati da Amazon S3 in Amazon Redshift utilizzando AWS Glue, eseguendo operazioni di estrazione, trasformazione e caricamento (ETL).

I file sorgente in Amazon S3 possono avere diversi formati, tra cui file con valori separati da virgole (CSV), XML e JSON. Questo modello descrive come utilizzare AWS Glue per convertire i file sorgente in un formato ottimizzato in termini di costi e prestazioni come Apache Parquet. Puoi interrogare i file Parquet direttamente da Amazon Athena e Amazon Redshift Spectrum. Puoi anche caricare file Parquet in Amazon Redshift, aggregarli e condividere i dati aggregati con i consumatori o visualizzare i dati utilizzando Amazon Quick Sight.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un bucket sorgente S3 con i privilegi giusti e contenente file CSV, XML o JSON.

Ipotesi

- I file sorgente CSV, XML o JSON sono già caricati in Amazon S3 e sono accessibili dall'account in cui sono configurati AWS Glue e Amazon Redshift.
- Vengono seguite le best practice per il caricamento dei file, la suddivisione dei file, la compressione e l'utilizzo di un manifesto, come illustrato nella documentazione di Amazon Redshift.
- La struttura del file di origine è inalterata.
- Il sistema di origine è in grado di importare dati in Amazon S3 seguendo la struttura delle cartelle definita in Amazon S3.

- Il cluster Amazon Redshift si estende su una singola zona di disponibilità. (Questa architettura è appropriata perché AWS Lambda, AWS Glue e Amazon Athena sono serverless.) Per un'elevata disponibilità, le istantanee del cluster vengono scattate con una frequenza regolare.

Limitazioni

- I formati di file sono limitati a quelli attualmente supportati da AWS Glue.
- La reportistica downstream in tempo reale non è supportata.

Architecture

Stack tecnologico di origine

- Bucket S3 con file CSV, XML o JSON

Stack tecnologico Target

- Data lake S3 (con archiviazione di file Parquet partizionata)
- Amazon Redshift

Architettura Target

Flusso di dati

Tools (Strumenti)

- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti altamente scalabile. Amazon S3 può essere utilizzato per un'ampia gamma di soluzioni di storage, tra cui siti Web, applicazioni mobili, backup e data lake.
- AWS Lambda: AWS Lambda consente di eseguire codice senza effettuare il provisioning o la gestione di server. AWS Lambda è un servizio basato sugli eventi; puoi configurare il codice per l'avvio automatico da altri servizi AWS.

- Amazon Redshift — Amazon Redshift è un servizio di data warehouse completamente gestito su scala petabyte. Con Amazon Redshift, puoi interrogare petabyte di dati strutturati e semistrutturati nel tuo data warehouse e nel tuo data lake utilizzando SQL standard.
- AWS Glue — AWS Glue è un servizio ETL completamente gestito che semplifica la preparazione e il caricamento dei dati per l'analisi. AWS Glue rileva i tuoi dati e archivia i metadati associati (ad esempio, definizioni di tabelle e schemi) nel catalogo dati di AWS Glue. I dati catalogati sono immediatamente ricercabili, possono essere interrogati e sono disponibili per ETL.
- AWS Secrets Manager: AWS Secrets Manager facilita la protezione e la gestione centralizzata dei segreti necessari per l'accesso alle applicazioni o ai servizi. Il servizio archivia le credenziali del database, le chiavi API e altri segreti ed elimina la necessità di codificare le informazioni sensibili in formato testo semplice. Secrets Manager offre anche la rotazione delle chiavi per soddisfare le esigenze di sicurezza e conformità. È dotato di integrazione integrata per Amazon Redshift, Amazon Relational Database Service (Amazon RDS) e Amazon DocumentDB. È possibile archiviare e gestire centralmente i segreti utilizzando la console Secrets Manager, l'interfaccia a riga di comando (CLI) o l'API Secrets Manager e. SDKs
- Amazon Athena — Amazon Athena è un servizio di query interattivo che semplifica l'analisi dei dati archiviati in Amazon S3. Athena è serverless e integrata con AWS Glue, quindi può interrogare direttamente i dati catalogati utilizzando AWS Glue. Athena è scalabile in modo elastico per offrire prestazioni di query interattive.

Epiche

Crea i bucket S3 e la struttura delle cartelle

Operazione	Description	Competenze richieste
Analizza i sistemi di origine per la struttura e gli attributi dei dati.	Esegui questa attività per ogni fonte di dati che contribuisce al data lake Amazon S3.	Ingegnere dei dati
Definisci la strategia di partizione e accesso.	Questa strategia dovrebbe basarsi sulla frequenza dell'acquisizione dei dati, sull'elaborazione delta e sulle esigenze di consumo. Assicurati che i bucket S3	Ingegnere dei dati

Operazione	Description	Competenze richieste
	non siano aperti al pubblico e che l'accesso sia controllato solo da politiche specifiche basate sui ruoli di servizio. Per ulteriori informazioni, consulta la Documentazione di Amazon S3.	
Crea bucket S3 separati per ogni tipo di origine dati e un bucket S3 separato per origine per i dati elaborati (Parquet).	Crea un bucket separato per ogni fonte, quindi crea una struttura di cartelle basata sulla frequenza di inserimento dei dati del sistema di origine, ad esempio. `s3://source-system-name/date/hour` Per i file elaborati (convertiti in formato Parquet), create una struttura simile, ad esempio. `s3://source-processed-bucket/date/hour` Per ulteriori informazioni sulla creazione di bucket S3, consulta la documentazione di Amazon S3.	Ingegnere dei dati

Crea un data warehouse in Amazon Redshift

Operazione	Description	Competenze richieste
Avvia il cluster Amazon Redshift con i gruppi di parametri e la strategia di	Usa il segreto del database Secrets Manager per le credenziali degli utenti amministratori durante la	Ingegnere dei dati

Operazione	Description	Competenze richieste
manutenzione e backup appropriati.	creazione del cluster Amazon Redshift. Per informazioni sulla creazione e il dimensionamento di un cluster Amazon Redshift, consulta la documentazione di Amazon Redshift e il white paper di Sizing Cloud Data Warehouses.	
Crea e collega il ruolo del servizio IAM al cluster Amazon Redshift.	Il ruolo del servizio AWS Identity and Access Management (IAM) garantisce l'accesso a Secrets Manager e ai bucket S3 di origine. Per ulteriori informazioni, consulta la documentazione AWS sull'autorizzazione e l'aggiunta di un ruolo.	Ingegnere dei dati
Crea lo schema del database.	Segui le best practice di Amazon Redshift per la progettazione di tabelle. In base al caso d'uso, scegli le chiavi di ordinamento e distribuzione appropriate e la migliore codifica di compressione possibile. Per le best practice, consulta la documentazione di AWS.	Ingegnere dei dati

Operazione	Description	Competenze richieste
Configura la gestione del carico di lavoro.	Configura le code di gestione del carico di lavoro (WLM), l'accelerazione delle query brevi (SQA) o la scalabilità simultanea, a seconda delle tue esigenze. Per ulteriori informazioni, consulta Implementazione della gestione del carico di lavoro nella documentazione di Amazon Redshift.	Ingegnere dei dati

Creazione di un segreto in Secrets Manager

Operazione	Description	Competenze richieste
Crea un nuovo segreto per archiviare le credenziali di accesso di Amazon Redshift in Secrets Manager.	Questo segreto memorizza le credenziali per l'utente amministratore e per i singoli utenti del servizio di database. Per istruzioni, consulta la documentazione di Secrets Manager. Scegli Amazon Redshift Cluster come tipo segreto. Inoltre, nella pagina di rotazione segreta, attiva la rotazione. Questo creerà l'utente appropriato nel cluster Amazon Redshift e ruoterà i segreti chiave a intervalli definiti.	Ingegnere dei dati

Operazione	Description	Competenze richieste
Crea una policy IAM per limitare l'accesso a Secrets Manager.	Limita l'accesso a Secrets Manager solo agli amministratori di Amazon Redshift e AWS Glue.	Ingegnere dei dati

Configura AWS Glue

Operazione	Description	Competenze richieste
Nel catalogo dati di AWS Glue, aggiungi una connessione per Amazon Redshift.	Per istruzioni, consulta la documentazione di AWS Glue.	Ingegnere dei dati
Crea e associa un ruolo di servizio IAM per AWS Glue per accedere a Secrets Manager, Amazon Redshift e bucket S3.	Per ulteriori informazioni, consulta la documentazione di AWS Glue.	Ingegnere dei dati
Definisci il catalogo dati di AWS Glue per l'origine.	Questa fase prevede la creazione di un database e delle tabelle obbligatorie nel catalogo dati di AWS Glue. Puoi utilizzare un crawler per catalogare le tabelle nel database AWS Glue o definirle come tabelle esterne Amazon Athena. Puoi anche accedere alle tabelle esterne definite in Athena tramite AWS Glue Data Catalog. Consulta la documentazione di AWS per ulteriori informazioni sulla definizione del Data Catalog e	Ingegnere dei dati

Operazione	Description	Competenze richieste
	sulla creazione di una tabella esterna in Athena.	
Crea un job AWS Glue per elaborare i dati di origine.	Il job AWS Glue può essere usato come shell Python o PySpark per standardizzare, deduplicare e pulire i file dei dati di origine. Per ottimizza re le prestazioni ed evitare di dover interrogare l'intero bucket di origine S3, partizion a il bucket S3 per data, suddiviso per anno, mese, giorno e ora come predicato pushdown per il job AWS Glue. Per ulteriori informazi oni, consulta la documenta zione di AWS Glue. Carica i dati elaborati e trasformati nelle partizioni del bucket S3 elaborate in formato Parquet. È possibile interrogare i file Parquet da Athena.	Ingegnere dei dati
Crea un job AWS Glue per caricare dati in Amazon Redshift.	Il job AWS Glue può essere una shell Python o PySpark caricare i dati invertendo i dati, seguito da un aggiornam ento completo. Per i dettagli, consulta la documentazione di AWS Glue e la sezione Informazioni aggiuntive.	Ingegnere dei dati

Operazione	Description	Competenze richieste
(Facoltativo) Pianifica i lavori AWS Glue utilizzando i trigger, se necessario.	Il carico di dati incrementale è principalmente guidato da un evento Amazon S3 che fa sì che una funzione AWS Lambda chiami il job AWS Glue. Utilizza la pianificazione basata su trigger di AWS Glue per tutti i carichi di dati che richiedono una pianificazione basata sul tempo anziché una pianificazione basata sugli eventi.	Ingegnere dei dati

Creazione di una funzione Lambda

Operazione	Description	Competenze richieste
Crea e collega un ruolo collegato ai servizi IAM per AWS Lambda per accedere ai bucket S3 e al job AWS Glue.	Crea un ruolo collegato ai servizi IAM per AWS Lambda con una policy per leggere oggetti e bucket Amazon S3 e una policy per accedere all'API AWS Glue per avviare un job AWS Glue. Per ulteriori informazioni, consulta il Knowledge Center.	Ingegnere dei dati
Crea una funzione Lambda per eseguire il job AWS Glue in base all'evento Amazon S3 definito.	La funzione Lambda deve essere avviata con la creazione del file manifest di Amazon S3. La funzione Lambda deve passare la posizione della cartella Amazon S3 (ad esempio,	Ingegnere dei dati

Operazione	Description	Competenze richieste
	bucket/year/month/date/ hour source_) al job AWS Glue come parametro. Il job AWS Glue utilizzerà questo parametro come predicato pushdown per ottimizzare l'accesso ai file e le prestazioni di elaborazione dei lavori. Per ulteriori informazioni, consulta la documentazione di AWS Glue.	
Crea un evento oggetto Amazon S3 PUT per rilevare la creazione di oggetti e chiama la rispettiva funzione Lambda.	L'evento oggetto PUT di Amazon S3 deve essere avviato solo con la creazione del file manifest. Il file manifest controlla la funzione Lambda e la concorrenza dei job AWS Glue ed elabora il carico come batch invece di elaborare singoli file che arrivano in una partizione specifica del bucket di origine S3. Per ulteriori informazioni, consulta la documentazione di Lambda.	Ingegnere dei dati

Risorse correlate

- Documentazione Amazon S3
- Documentazione AWS Glue
- Documentazione Amazon Redshift
- AWS Lambda
- Amazon Athena

- AWS Secrets Manager

Informazioni aggiuntive

Approccio dettagliato per una modifica e un aggiornamento completo

Upsert: è destinato ai set di dati che richiedono l'aggregazione storica, a seconda del caso d'uso aziendale. Segui uno degli approcci descritti in Aggiornamento e inserimento di nuovi dati (documentazione di Amazon Redshift) in base alle tue esigenze aziendali.

Aggiornamento completo: questo è per piccoli set di dati che non necessitano di aggregazioni storiche. Segui uno di questi approcci:

1. Tronca la tabella Amazon Redshift.
2. Carica la partizione corrente dall'area di staging

oppure:

1. Crea una tabella temporanea con i dati della partizione corrente.
2. Elimina la tabella Amazon Redshift di destinazione.
3. Rinomina la tabella temporanea nella tabella di destinazione.

Crea una rete dati aziendale con Amazon DataZone e AWS CDK AWS CloudFormation

Dhrubajyoti Mukherjee, Ravi Kumar, Weizhou Sun e Adjoa Taylor, Amazon Web Services

Riepilogo

Su Amazon Web Services (AWS), i clienti comprendono che i dati sono la chiave per accelerare l'innovazione e creare valore aziendale. Per gestire questa enorme quantità di dati, puoi adottare un'architettura decentralizzata come la data mesh. Un'architettura data mesh facilita la riflessione sul prodotto, una mentalità che tiene conto dei clienti, degli obiettivi e del mercato. La data mesh aiuta anche a stabilire un modello di governance federato che fornisce un accesso rapido e sicuro ai dati.

In Strategie per la creazione di una soluzione aziendale basata su data mesh viene AWS illustrato come utilizzare il Data Mesh Strategy Framework per formulare e implementare una strategia di data mesh per l'organizzazione. Utilizzando il Data Mesh Strategy Framework, puoi ottimizzare l'organizzazione dei team e le loro interazioni per accelerare il percorso verso la data mesh.

Questo documento fornisce indicazioni su come creare una rete dati aziendale con Amazon DataZone. Amazon DataZone è un servizio di gestione dei dati per la catalogazione, la scoperta, la condivisione e la gestione dei dati archiviati su fonti AWS, locali e di terze parti. Il modello include artefatti di codice che aiutano a implementare l'infrastruttura di soluzioni di dati basata su data mesh utilizzando e. AWS Cloud Development Kit (AWS CDK) AWS CloudFormation Questo modello è destinato agli architetti e agli ingegneri del cloud. DevOps

Per informazioni sugli obiettivi di questo modello e sull'ambito della soluzione, consulta la sezione Informazioni aggiuntive.

Prerequisiti e limitazioni

Prerequisiti

- Almeno due account attivi Account AWS: uno per l'account di governance centrale e l'altro per l'account del membro
- AWS credenziali di amministratore per l'account di governance centrale nell'ambiente di sviluppo
- AWS Command Line Interface (AWS CLI) installato per gestirlo Servizi AWS dalla riga di comando
- Node.js e Node Package Manager (npm) installati per gestire le applicazioni AWS CDK

- AWS CDK Toolkit installato a livello globale nell'ambiente di sviluppo utilizzando npm, per sintetizzare e distribuire applicazioni AWS CDK

```
npm install -g aws-cdk
```

- Python versione 3.12 installata nel tuo ambiente di sviluppo
- TypeScript installato nel tuo ambiente di sviluppo o installato globalmente utilizzando il compilatore npm:

```
npm install -g typescript
```

- Docker installato nel tuo ambiente di sviluppo
- Un sistema di controllo della versione come Git per mantenere il codice sorgente della soluzione (consigliato)
- Un ambiente di sviluppo integrato (IDE) o un editor di testo con supporto per Python e TypeScript (fortemente consigliato)

Limitazioni

- La soluzione è stata testata solo su macchine che eseguono Linux o macOS.
- Nella versione attuale, la soluzione non supporta l'integrazione di Amazon DataZone e AWS IAM Identity Center per impostazione predefinita. Tuttavia, puoi configurarla per supportare questa integrazione.

Versioni del prodotto

- Python versione 3.12

Architecture

Il diagramma seguente mostra un'architettura di riferimento per la rete di dati. L'architettura è basata su Amazon DataZone e utilizza Amazon Simple Storage Service (Amazon S3) AWS Glue Data Catalog e come fonti di dati. L'uso Servizi AWS che utilizzi con Amazon DataZone nell'implementazione della rete di dati potrebbe differire in base ai requisiti della tua organizzazione.

1. Nei conti dei produttori, i dati grezzi sono idonei al consumo nella loro forma attuale oppure vengono trasformati per il consumo utilizzando AWS Glue. I metadati tecnici per i dati vengono archiviati in Amazon S3 e valutati utilizzando AWS Glue un crawler di dati. La qualità dei dati viene misurata utilizzando Data Quality.AWS Glue Il database di origine nel catalogo dati è registrato come risorsa nel DataZone catalogo Amazon. Il DataZone catalogo Amazon è ospitato nell'account di governance centrale utilizzando Amazon DataZone Data Source Job.
2. L'account di governance centrale ospita il DataZone dominio Amazon e il portale DataZone dati Amazon. I produttori e i consumatori Account AWS di dati sono associati al DataZone dominio Amazon. I DataZone progetti Amazon dei produttori e dei consumatori di dati sono organizzati nelle corrispondenti unità di DataZone dominio Amazon.
3. Gli utenti finali degli asset di dati accedono al portale DataZone dati Amazon utilizzando le proprie credenziali AWS Identity and Access Management (IAM) o Single Sign-On (con integrazione tramite IAM Identity Center). Cercano, filtrano e visualizzano le informazioni sugli asset (ad esempio, informazioni sulla qualità dei dati o metadati aziendali e tecnici) nel catalogo DataZone dati di Amazon.
4. Dopo che un utente finale trova la risorsa di dati che desidera, utilizza la funzionalità di DataZone abbonamento Amazon per richiedere l'accesso. Il proprietario dei dati del team di produzione riceve una notifica e valuta la richiesta di abbonamento nel portale DataZone dati di Amazon. Il proprietario dei dati approva o rifiuta la richiesta di abbonamento in base alla sua validità.
5. Dopo che la richiesta di abbonamento è stata concessa e soddisfatta, si accede alla risorsa nell'account del consumatore per le seguenti attività:
   - Sviluppo di modelli AI/ML utilizzando Amazon AI SageMaker
   - Analisi e reportistica utilizzando Amazon Athena e Amazon Quick

Tools (Strumenti)

Servizi AWS

- Amazon Athena è un servizio di query interattivo che ti aiuta ad analizzare i dati direttamente in Amazon Simple Storage Service (Amazon S3) utilizzando SQL standard.
- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che ti aiuta a definire e fornire l' Cloud AWS infrastruttura in codice.
- AWS CloudFormationti aiuta a configurare AWS le risorse, a fornirle in modo rapido e coerente e a gestirle durante tutto il loro ciclo di vita tra Account AWS e. Regioni AWS

- Amazon DataZone è un servizio di gestione dei dati che ti aiuta a catalogare, scoprire, condividere e gestire i dati archiviati su AWS, in sede e in fonti di terze parti.
- Amazon Quick è un servizio di business intelligence (BI) su scala cloud che ti aiuta a visualizzare, analizzare e generare report sui dati in un'unica dashboard.
- Amazon SageMaker AI è un servizio di machine learning (ML) gestito che ti aiuta a creare e addestrare modelli di machine learning per poi distribuirli in un ambiente ospitato pronto per la produzione.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Simple Queue Service (Amazon SQS) fornisce una coda ospitata sicura, durevole e disponibile che ti aiuta a integrare e disaccoppiare sistemi e componenti software distribuiti.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Repository di codice

La soluzione è disponibile nel repository GitHub data-mesh-datazone-cdk-cloudformation.

Epiche

Configura l'ambiente

Operazione	Description	Competenze richieste
Clonare il repository.	Per clonare il repository, esegui il seguente comando nel tuo ambiente di sviluppo locale (Linux o macOS): `git clone https://github.com/aws-samples/data-mesh-datazone-cdk-cloudformation`	Architetto del cloud, ingegnere DevOps

Operazione	Description	Competenze richieste

Creazione dell'ambiente	Per creare l'ambiente virtuale Python, esegui i seguenti comandi:
	`python3 -m venv .venv`
	`source .venv/bin/activate`
	`pip install -r requirements.txt`
		Architetto del cloud, DevOps ingegnere

Avvia l'account.	Per avviare l'account di governance centrale utilizzando AWS CDK, esegui il seguente comando: `cdk bootstrap aws://<GOVERNANCE_ACCOUNT_ID>/<AWS_REGION>` Accedi a Console di gestione AWS, apri la console dell'account di governance centrale e ottieni l'Amazon Resource Name (ARN) del ruolo di AWS CDK esecuzione.	Architetto del cloud, DevOps ingegnere
Costruisci il `DzDataMeshMemberStackSet.yaml` file.	Per costruire il `DzDataMeshMemberStackSet.yaml` file, dalla directory principale del repository, iniziate il seguente script bash: `./lib/scripts/create_dz_data_mesh_member_stack_set.sh`	Architetto del cloud, ingegnere DevOps

Operazione	Description	Competenze richieste
Conferma la creazione del modello.	Assicuratevi che il file CloudFormation modello sia stato creato nella `lib/cfn-templates/DzDataMeshMemberStackSet.yaml` posizione desiderata.	Architetto del cloud, DevOps ingegnere

Implementa le risorse nell'account di governance centrale

Operazione	Description	Competenze richieste

Modifica la configurazione.	Nel `config/Config.ts` file, modificate i seguenti parametri:
	`DZ_APPLICATION_NAME - Name of the application.`
	`DZ_STAGE_NAME - Name of the stage.`
	`DZ_DOMAIN_NAME - Name of the Amazon DataZone domain`
	`DZ_DOMAIN_DESCRIPTION - Description of the Amazon DataZone domain`
	`DZ_DOMAIN_TAG - Tag of the Amazon DataZone domain`
	`DZ_ADMIN_PROJECT_NAME - Name of the Amazon DataZone project for administrators`
	`DZ_ADMIN_PROJECT_DESCRIPTION - Description of the Amazon DataZone project for administrators`
		Architetto del cloud, DevOps ingegnere

Operazione	Description	Competenze richieste

	`CDK_EXEC_ROLE_ARN - ARN of the cdk execution role`
	`DZ_ADMIN_ROLE_ARN - ARN of the administrator role`
	
	Mantieni vuoti i parametri rimanenti.	

Operazione	Description	Competenze richieste
Aggiorna la configurazione del DataZone glossario Amazon.	Per aggiornare la configura zione del DataZone glossario Amazon nel `lib/utils/glossary_config.json` file, usa la seguente configura zione di esempio: `{ "GlossaryName": "PII Data", "GlossaryDescription": "If data source contains PII attributes", "GlossaryTerms": [{ "Name": "Yes", "ShortDescription": "Yes", "LongDescription": "Yes Glossary Term" }, { "Name": "No", "ShortDescription": "No", "LongDescription": "No Glossary Term" } ] }`	Architetto del cloud, DevOps ingegnere

Operazione	Description	Competenze richieste
Aggiorna la configurazione del modulo di DataZone metadati Amazon.	Per aggiornare la configurazione del modulo di DataZone metadati Amazon in`lib/utils/metadata_form_config.json file`, utilizza la seguente configurazione di esempio: `{ "FormName": "ScheduleDataRefresh", "FormDescription": "Form for data refresh schedule", "FormSmithyModel": "@amazon.datazone#displayname(defaultName: \"Data Refresh Schedule\")\nstructure ScheduleDataRefresh {\n @documentation(\"Schedule of Data Refresh\")\n @required\n @amazon.datazone#searchable\n @amazon.datazone#displayname(defaultName: \"Data Refresh Schedule\")\n data_refresh_schedule: String\n}" }`	Architetto del cloud, DevOps ingegnere

Operazione	Description	Competenze richieste
Esporta le AWS credenziali.	Per esportare AWS le credenziali nel tuo ambiente di sviluppo per il ruolo IAM con autorizzazioni amministrative, utilizza il seguente formato: `export AWS_ACCES S_KEY_ID= export AWS_SECRE T_ACCESS_KEY= export AWS_SESSI ON_TOKEN=`	Architetto del cloud, ingegnere DevOps
Sintetizza il modello.	Per sintetizzare il CloudForm ation modello, esegui il seguente comando: `npx cdk synth`	Architetto del cloud, ingegnere DevOps
Distribuire la soluzione.	Per distribuire la soluzione, esegui il comando seguente: `npx cdk deploy --all`	Architetto del cloud, DevOps ingegnere

Configura nuovi account per i membri

Operazione	Description	Competenze richieste
Implementa il modello.	Distribuisci il CloudForm ation modello che si trova `lib/cfn-templates/ DzDataMeshCfnStack SetExecutionRole.y aml` nell'account del membro	Architetto del cloud, ingegnere DevOps

Operazione	Description	Competenze richieste
	con i seguenti parametri di input: • `GovernanceAccountID` – ID dell'account di governance • `DataZoneKMSKeyID` – ID della chiave AWS Key Management Service (AWS KMS) che crittografa i metadati di Amazon DataZone • `NotificationQueueName` – Nome della coda di notifica di Amazon SQS nell'account di governance	
Aggiorna il ARNs.	Per aggiornare l'elenco dei ruoli di CloudFormation StackSet esecuzione ARNs per gli account membri, utilizza il codice seguente: `DZ_MEMBER_STACK_SET_EXEC_ROLE_LIST - List of Stack set execution role arns for the member accounts.`	Architetto del cloud, DevOps ingegnere

Operazione	Description	Competenze richieste

Sintetizza e distribuisci.	Per sintetizzare il CloudFormation modello e distribuire la soluzione, esegui i seguenti comandi:
	`npx cdk synth`
	`npx cdk deploy --all`
		Architetto del cloud, ingegnere DevOps

Operazione	Description	Competenze richieste
Associa l'account membro.	Per associare l'account membro all'account di governance centrale, procedi come segue: 1. Accedi alla console dell'account di governance centrale e apri la DataZone console Amazon all'indirizzo https://console.aws.amazon.com/datazone/. 2. Scegli il dominio che hai creato. 3. Scorri fino alla scheda Account associati e scegli Richiedi associazione. 4. Fornisci l'ID dell'account AWS e scegli AWSRAMPermissionDataZonePortalReadWritecome policy RAM. 5. Scegli Request Association. 6. Attendi di ricevere una notifica via email che ti comunica che il tuo account è stato avviato correttamente.	Architetto del cloud, ingegnere DevOps

Operazione	Description	Competenze richieste
Aggiorna i parametri.	Per aggiornare i parametri specifici dell'account membro nel file di configurazione in `config/Config.ts` , utilizza il seguente formato: `export const DZ_MEMBER_ACCOUNT_CONFIG: memberAccountConfig = { '123456789012' : { PROJECT_NAME: 'TEST-PROJECT-123456789012', PROJECT_DESCRIPTION: 'TEST-PROJECT-123456789012', PROJECT_EMAIL: 'user@xyz.com' } }`	Architetto del cloud, ingegnere DevOps

Sintetizza e implementa il modello.	Per sintetizzare il CloudFormation modello e distribuire la soluzione, esegui i comandi seguenti:
	`npx cdk synth`
	`npx cdk deploy --all`
		Architetto del cloud, ingegnere DevOps

Operazione	Description	Competenze richieste
Aggiungi account per i membri.	Per creare e configurare account membro aggiuntiv i nella soluzione dati, ripeti i passaggi precedenti per ogni account membro. Questa soluzione non fa distinzione tra produttori di dati e consumatori.	Architetto del cloud, DevOps ingegnere

Eliminazione

Operazione	Description	Competenze richieste
Dissocia gli account dei membri.	Per dissociare gli account, procedi come segue: 1. Accedi alla console e apri la DataZone console Amazon. 2. Scegli Visualizza domini. 3. Seleziona il dominio che hai creato. 4. Scegli la scheda Associazi oni di account. 5. Seleziona l'account membro da cui desideri dissociare. 6. Scegli Dissocia e inserisci `disassociate` per confermare. 7. Ripeti i passaggi 3-6 per tutti gli account dei membri.	Architetto del cloud, DevOps ingegnere

Operazione	Description	Competenze richieste
Elimina le istanze dello stack.	Per eliminare le istanze dello CloudFormation stack, procedi come segue: 1. Apri la CloudFormation console all'indirizzo. https://console.aws.amazon.com/cloudformation/ 2. Nel pannello di navigazione, scegli StackSets. 3. Scegliete lo stack set denominato StackSet-DataZone - DataMesh - Member e scegliete la scheda Stack instances. 4. Copia l' Account AWS ID dell'account membro che desideri rimuovere dall'iscrizione. 5. Scegli Azioni, scegli Elimina pile da StackSet e mantieni le opzioni predefinite. 6. Nel campo Numeri di account, inserisci l'ID dell'account. 7. Nell'elenco a discesa Specificare le regioni, scegli. Regione AWS 8. Scegli Avanti, quindi scegli Invia. 9. Nella scheda Operazioni, conferma che l'operazione è riuscita. L'elimina	Architetto del cloud, DevOps ingegnere

Operazione	Description	Competenze richieste
	zione dello stack potrebbe richiedere del tempo. 10. Ripeti i passaggi da 2 a 9 per tutti gli account dei membri.	
Distruggi tutte le risorse.	Per distruggere le risorse, implementa i seguenti passaggi nel tuo ambiente di sviluppo locale (Linux o macOS): 1. Passa alla directory principale del tuo repository. 2. Esporta AWS le credenziali per il ruolo IAM che hai usato per creare lo AWS CDK stack. 3. Per distruggere le risorse cloud, esegui il seguente comando: `npx cdk destroy --all`	Architetto del cloud, DevOps ingegnere

Risorse correlate

- Amazon DataZone quickstart con i dati AWS Glue
- Tutorial: crea la tua prima app AWS CDK
- Iniziare con CloudFormation
- Strategie per creare una soluzione aziendale basata su data mesh su AWS

Informazioni aggiuntive

Obiettivi

L'implementazione di questo modello consente di ottenere quanto segue:

- Proprietà decentralizzata dei dati – Trasferisci la proprietà dei dati da un team centrale ai team che rappresentano i sistemi di origine, le unità aziendali o i casi d'uso dell'organizzazione.
- Pensiero di prodotto – Introduci una mentalità basata sul prodotto che includa i clienti, il mercato e altri fattori quando consideri le risorse di dati della tua organizzazione.
- Governance federata – Migliora le barriere di sicurezza, i controlli e la conformità tra i prodotti di dati dell'organizzazione.
- Supporto per più account e più progetti – Supporta la condivisione e la collaborazione dei dati efficienti e sicure tra le unità aziendali o i progetti dell'organizzazione.
- Monitoraggio e notifiche centralizzati – Monitora le risorse cloud della tua mesh di dati utilizzando Amazon CloudWatch e avvisa gli utenti quando viene associato un nuovo account membro.
- Scalabilità ed estensibilità – Aggiungi nuovi casi d'uso alla data mesh man mano che la tua organizzazione si evolve.

Ambito della soluzione

Quando utilizzi questa soluzione, puoi iniziare con dimensioni ridotte e scalare man mano che avanzi nel percorso verso la rete di dati. Spesso, quando un account membro adotta la soluzione dati, contiene configurazioni di account specifiche per l'organizzazione, il progetto o l'unità aziendale. Questa soluzione si adatta a queste diverse Account AWS configurazioni supportando le seguenti funzionalità:

- AWS Glue Data Catalog come fonte di dati per Amazon DataZone
- Gestione del dominio DataZone dati Amazon e del relativo portale dati
- Gestione dell'aggiunta di account membri nella soluzione di dati basata su data mesh
- Gestione di DataZone progetti e ambienti Amazon
- Gestione di DataZone glossari e moduli di metadati Amazon
- Gestione dei ruoli IAM che corrispondono agli utenti della soluzione di dati basata su data mesh
- Notifica agli utenti di soluzioni di dati basate su data mesh
- Monitoraggio dell'infrastruttura cloud predisposta

Questa soluzione utilizza AWS CDK e CloudFormation implementa l'infrastruttura cloud. Viene utilizzata CloudFormation per eseguire le seguenti operazioni:

- Definisci e distribuisci le risorse cloud a un livello di astrazione inferiore.
- Implementa risorse cloud da. Console di gestione AWS Utilizzando questo approccio, è possibile implementare l'infrastruttura senza un ambiente di sviluppo.

La soluzione data mesh consente di definire le risorse AWS CDK a un livello di astrazione più elevato. Di conseguenza, la soluzione fornisce un approccio disaccoppiato, modulare e scalabile scegliendo lo strumento pertinente per implementare le risorse cloud.

Fasi successive

Puoi rivolgerti agli AWSesperti per ricevere indicazioni su come creare una rete di dati con Amazon DataZone.

La natura modulare di questa soluzione supporta la creazione di soluzioni di gestione dei dati con diverse architetture, come data fabric e data lake. Inoltre, in base ai requisiti della tua organizzazione, puoi estendere la soluzione ad altre fonti di DataZone dati Amazon.

Calcola il valore a rischio (VaR) utilizzando i servizi AWS

Sumon Samanta, Amazon Web Services

Riepilogo

Questo modello descrive come implementare un sistema di calcolo del valore a rischio (VaR) utilizzando i servizi AWS. In un ambiente locale, la maggior parte dei sistemi VaR utilizza un'ampia infrastruttura dedicata e un software di pianificazione della rete interno o commerciale per eseguire processi in batch. Questo modello presenta un'architettura semplice, affidabile e scalabile per gestire l'elaborazione VaR nel cloud AWS. Crea un'architettura serverless che utilizza Amazon Kinesis Data Streams come servizio di streaming, Amazon Simple Queue Service (Amazon SQS) come servizio di coda gestito, Amazon come servizio di cache e AWS Lambda per elaborare ElastiCache gli ordini e calcolare il rischio.

Il VaR è una misura statistica che i trader e i gestori del rischio utilizzano per stimare le potenziali perdite nel loro portafoglio oltre un certo livello di confidenza. La maggior parte dei sistemi VaR prevede l'esecuzione di un gran numero di calcoli matematici e statistici e l'archiviazione dei risultati. Questi calcoli richiedono risorse di calcolo significative, quindi i processi batch VaR devono essere suddivisi in set più piccoli di attività di calcolo. La suddivisione di un batch di grandi dimensioni in attività più piccole è possibile perché queste attività sono per lo più indipendenti (ovvero, i calcoli per un'attività non dipendono da altre attività).

Un altro requisito importante per un'architettura VaR è la scalabilità di calcolo. Questo modello utilizza un'architettura serverless che si ridimensiona automaticamente in avanti o indietro in base al carico di calcolo. Poiché la domanda di elaborazione in batch o online è difficile da prevedere, è necessaria la scalabilità dinamica per completare il processo entro la tempistica imposta da un accordo sul livello di servizio (SLA). Inoltre, un'architettura ottimizzata in termini di costi dovrebbe essere in grado di ridimensionare ogni risorsa di elaborazione non appena le attività su tale risorsa sono complete.

I servizi AWS sono adatti per i calcoli VaR perché offrono capacità di calcolo e storage scalabile, servizi di analisi per l'elaborazione in modo ottimizzato in termini di costi e diversi tipi di scheduler per eseguire i flussi di lavoro di gestione del rischio. Inoltre, paghi solo per le risorse di calcolo e storage che usi su AWS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- File di input, che dipendono dai requisiti aziendali. Un tipico caso d'uso riguarda i seguenti file di input:
  - File di dati di mercato (input nel motore di calcolo del VaR)
  - File di dati commerciali (a meno che i dati commerciali non arrivino attraverso un flusso).
  - File di dati di configurazione (modello e altri dati di configurazione statici)
  - File di modello del motore di calcolo (librerie quantitative)
  - File di dati delle serie temporali (per dati storici come il prezzo delle azioni degli ultimi cinque anni)
- Se i dati di mercato o altri input arrivano tramite un flusso, vengono configurate le autorizzazioni di Amazon Kinesis Data Streams e Amazon Identity and Access Management (IAM) configurate per scrivere nello stream.

Questo modello crea un'architettura in cui i dati commerciali vengono scritti da un sistema di trading a un flusso di dati Kinesis. Invece di utilizzare un servizio di streaming, puoi salvare i dati commerciali in piccoli file batch, archiviarli in un bucket Amazon Simple Storage Service (Amazon S3) e richiamare un evento per avviare l'elaborazione dei dati.

Limitazioni

- Il sequenziamento del flusso di dati Kinesis è garantito su ogni shard, pertanto non è garantito che gli ordini commerciali scritti su più shard vengano consegnati nello stesso ordine delle operazioni di scrittura.
- Il limite di runtime di AWS Lambda è attualmente di 15 minuti. (Per ulteriori informazioni, consulta le domande frequenti su Lambda).

Architecture

Architettura Target

Il seguente diagramma di architettura mostra i servizi e i flussi di lavoro AWS per il sistema di valutazione del rischio.

Il diagramma illustra quanto segue:

1. Le negoziazioni arrivano dal sistema di gestione degli ordini.
2. La funzione Lambda di ticket position netting elabora gli ordini e scrive messaggi consolidati per ogni ticker in una coda di rischio in Amazon SQS.
3. La funzione Lambda del motore di calcolo del rischio elabora i messaggi di Amazon SQS, esegue calcoli del rischio e aggiorna le informazioni su profitti e perdite (PnL) del VaR nella cache dei rischi di Amazon. ElastiCache
4. La funzione Lambda di lettura ElastiCache dei dati recupera i risultati del rischio e li archivia in un database ElastiCache e in un bucket S3.

Per ulteriori informazioni su questi servizi e passaggi, consulta la sezione Epics.

Automazione e scalabilità

Puoi distribuire l'intera architettura utilizzando l'AWS Cloud Development Kit (AWS CDK) o i modelli CloudFormation AWS. L'architettura può supportare sia l'elaborazione in batch che l'elaborazione intraday (in tempo reale).

La scalabilità è integrata nell'architettura. Man mano che sempre più operazioni vengono scritte nel flusso di dati di Kinesis e sono in attesa di essere elaborate, è possibile richiamare funzioni Lambda aggiuntive per elaborare tali operazioni e ridurle al termine dell'elaborazione. Un'altra opzione è l'elaborazione tramite più code di calcolo del rischio di Amazon SQS. Se è richiesto un ordinamento o un consolidamento rigorosi tra le code, l'elaborazione non può essere parallelizzata. Tuttavia, per un end-of-the-day batch o un mini batch intraday, le funzioni Lambda possono elaborare in parallelo e memorizzare i risultati finali in. ElastiCache

Tools (Strumenti)

Servizi AWS

- Amazon Aurora MySQL Compatible Edition è un motore di database relazionale completamente gestito e compatibile con MySQL che ti aiuta a configurare, gestire e scalare le distribuzioni MySQL. Questo modello utilizza MySQL come esempio, ma è possibile utilizzare qualsiasi sistema RDBMS per archiviare i dati.
- Amazon ti ElastiCache aiuta a configurare, gestire e scalare ambienti di cache in memoria distribuiti nel cloud AWS.
- Amazon Kinesis Data Streams ti aiuta a raccogliere ed elaborare grandi flussi di record di dati in tempo reale.

- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- Amazon Simple Queue Service (Amazon SQS) fornisce una coda ospitata sicura, durevole e disponibile che ti aiuta a integrare e disaccoppiare sistemi e componenti software distribuiti.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Codice

Questo modello fornisce un'architettura di esempio per un sistema VaR nel cloud AWS e descrive come utilizzare le funzioni Lambda per i calcoli del VaR. Per creare le tue funzioni Lambda, consulta gli esempi di codice nella documentazione di Lambda. Per assistenza, contatta AWS Professional Services.

Best practice

- Mantieni ogni attività di calcolo VaR il più piccola e leggera possibile. Sperimenta un numero diverso di operazioni in ciascuna attività di calcolo per vedere quale è la più ottimizzata in termini di tempi e costi di calcolo.
- Archivia oggetti riutilizzabili in Amazon ElastiCache. Usa un framework come Apache Arrow per ridurre la serializzazione e la deserializzazione.
- Considera la limitazione temporale di Lambda. Se ritieni che le tue attività di elaborazione possano superare i 15 minuti, prova a suddividerle in attività più piccole per evitare il timeout Lambda. Se ciò non è possibile, potresti prendere in considerazione una soluzione di orchestrazione dei container con AWS Fargate, Amazon Elastic Container Service (Amazon ECS) e Amazon Elastic Kubernetes Service (Amazon EKS).

Epiche

Sistema dal flusso commerciale al rischio

Operazione	Description	Competenze richieste
Inizia a scrivere scambi.	Le negoziazioni nuove, liquidate o parzialmente	Amazon Kinesis

Operazione	Description	Competenze richieste
	regolate vengono registrate dal sistema di gestione degli ordini in un flusso di rischio. Questo modello utilizza Amazon Kinesis come servizio di streaming gestito. L'hash del trade order ticker viene utilizzato per inserire gli ordini commerciali su più frammenti.	

Esegui le funzioni Lambda per l'elaborazione degli ordini

Operazione	Description	Competenze richieste
Inizia l'elaborazione del rischio con Lambda.	Esegui una funzione AWS Lambda per i nuovi ordini. In base al numero di ordini commerciali in sospeso, Lambda si ridimensionerà automaticamente. Ogni istanza Lambda ha uno o più ordini e recupera la posizione più recente per ogni ticker da Amazon. ElastiCache (Puoi utilizzare un ID CUSIP, un nome Curve o un nome di indice per altri prodotti finanziari derivati come chiave per archiviare e recuperar e dati.) ElasticCache In ElastiCache, la posizione totale (quantità) e la coppia chiave-valore < ticker, net position >, dove la posizione	Amazon Kinesis, AWS Lambda, Amazon ElastiCache

Operazione	Description	Competenze richieste
	netta è il fattore di scala, vengono aggiornate una volta per ogni ticker.	

Scrivi messaggi per ogni ticker in coda

Operazione	Description	Competenze richieste
Scrivi messaggi consolidati nella coda di rischio.	Scrivi il messaggio in una coda. Questo modello utilizza Amazon SQS come servizio di coda gestito. Una singola istanza Lambda può ricevere un mini batch di ordini commerciali in qualsiasi momento, ma scriverà solo un messaggio per ogni ticker su Amazon SQS. Viene calcolato un fattore di scala: (vecchia posizione netta+posizione attuale)/vecchia posizione netta.	Amazon SQS, AWS Lambda

Richiama il motore di rischio

Operazione	Description	Competenze richieste
Avvia i calcoli del rischio.	Viene richiamata la funzione Lambda per il risk engine lambda. Ogni posizione viene elaborata da una singola funzione Lambda. Tuttavia, a fini di ottimizzazione, ogni funzione Lambda può	Amazon SQS, AWS Lambda

Operazione	Description	Competenze richieste
	elaborare più messaggi da Amazon SQS.	

Recupera i risultati del rischio dalla cache

Operazione	Description	Competenze richieste
Recupera e aggiorna la cache dei rischi.	Lambda recupera la posizione netta corrente per ogni ticker da. ElastiCache Inoltre, recupera un array di profitti e perdite (PnL) VaR per ogni ticker da. ElastiCache Se l'array PnL esiste già, la funzione Lambda aggiorna l'array e il vAR con una scala, che proviene dal messaggio Amazon SQS scritto dalla funzione netting Lambda. Se l'array PnL non è presente ElasticCache, vengono calcolati nuovi PnL e VaR utilizzando dati simulati sulla serie di prezzi dei ticker.	Amazon SQS, AWS Lambda, Amazon ElastiCache

Aggiorna i dati in Elastic Cache e archivia nel database

Operazione	Description	Competenze richieste
Memorizza i risultati dei rischi.	Dopo l'aggiornamento dei numeri VaR e PnL ElastiCac he, viene richiamata una nuova funzione Lambda	AWS Lambda, Amazon ElastiCache

Operazione	Description	Competenze richieste
	ogni cinque minuti. Questa funzione legge tutti i dati memorizzati ElastiCache e li archivia in un database Aurora compatibile con MySQL e in un bucket S3.	

Risorse correlate

- Struttura di Basilea VaR

Configura l'accesso tra account a un catalogo di dati AWS Glue condiviso utilizzando Amazon Athena

Denis Avdonin, Amazon Web Services

Riepilogo

Questo modello fornisce step-by-step istruzioni, inclusi esempi di policy di AWS Identity and Access Management (IAM), per configurare la condivisione tra account di un set di dati archiviato in un bucket Amazon Simple Storage Service (Amazon S3) utilizzando AWS Glue Data Catalog. Puoi archiviare il set di dati in un bucket S3. I metadati vengono raccolti da un crawler di AWS Glue e inseriti nel catalogo dati di AWS Glue. Il bucket S3 e il catalogo dati AWS Glue risiedono in un account AWS denominato account dati. Puoi fornire l'accesso ai principali IAM in un altro account AWS denominato account consumer. Gli utenti possono interrogare i dati nell'account consumer utilizzando il motore di query serverless di Amazon Athena.

Prerequisiti e limitazioni

Prerequisiti

- Due account AWS attivi
- Un bucket S3 in uno degli account AWS
- Motore Athena versione 2
- AWS Command Line Interface (AWS CLI), installata e configurata (o AWS per l'esecuzione di comandi CloudShell AWS CLI)

Versioni del prodotto

Questo modello funziona solo con il motore Athena versione 2 e il motore Athena versione 3. Ti consigliamo di eseguire l'aggiornamento alla versione 3 del motore Athena. Se non riesci a eseguire l'upgrade dalla versione 1 del motore Athena alla versione 3 del motore Athena, segui l'approccio descritto in Accesso multiaccount ad AWS Glue Data Catalog con Amazon Athena nel blog di AWS Big Data.

Architecture

Stack tecnologico Target

- Amazon Athena
- Amazon Simple Storage Service (Amazon S3)
- AWS Glue
- AWS Identity and Access Management (IAM)
- AWS Key Management Service (AWS KMS)

Il diagramma seguente mostra un'architettura che utilizza le autorizzazioni IAM per condividere i dati in un bucket S3 in un account AWS (account dati) con un altro account AWS (account consumer) tramite AWS Glue Data Catalog.

Il diagramma mostra il flusso di lavoro seguente:

1. La policy S3 bucket nell'account dati concede le autorizzazioni a un ruolo IAM nell'account consumer e al ruolo del servizio crawler AWS Glue nell'account dati.
2. La policy chiave di AWS KMS nell'account dati concede le autorizzazioni al ruolo IAM nell'account consumer e al ruolo del servizio crawler AWS Glue nell'account dati.
3. Il crawler AWS Glue nell'account dati rileva lo schema dei dati archiviati nel bucket S3.
4. La policy sulle risorse di AWS Glue Data Catalog nell'account dati consente l'accesso al ruolo IAM nell'account consumer.
5. Un utente crea un riferimento di catalogo denominato nell'account consumer utilizzando un comando CLI AWS.
6. Una policy IAM garantisce a un ruolo IAM nell'account consumer l'accesso alle risorse dell'account di dati. La policy di fiducia del ruolo IAM consente agli utenti dell'account consumer di assumere il ruolo IAM.
7. Un utente dell'account consumer assume il ruolo IAM e accede agli oggetti nel catalogo dati utilizzando le query SQL.
8. Il motore serverless Athena esegue le query SQL.

> **Note**
>
> Le migliori pratiche IAM consigliano di concedere le autorizzazioni a un ruolo IAM e di utilizzare la federazione delle identità.

Tools (Strumenti)

- Amazon Athena è un servizio di query interattivo che ti aiuta ad analizzare i dati direttamente in Amazon S3 utilizzando SQL standard.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Glue è un servizio di estrazione, trasformazione e caricamento (ETL) completamente gestito. Ti aiuta a classificare, pulire, arricchire e spostare i dati in modo affidabile tra archivi di dati e flussi di dati.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.

Epiche

Imposta le autorizzazioni nell'account dati

Operazione	Description	Competenze richieste
Concedi l'accesso ai dati nel bucket S3.	Crea una policy per i bucket S3 basata sul modello seguente e assegna la policy al bucket in cui sono archiviati i dati. `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "AWS": [ "arn:aws:iam::<con`	Amministratore del cloud

Operazione	Description	Competenze richieste

	sumer account id>:role/
	<role name>",
	
	"arn:aws:iam::<dat
	a account id>:role/
	service-role/AWSGl
	ueServiceRole-data-
	bucket-crawler"
	]
	},
	"Action":
	"s3:GetObject",
	"Resource":
	"arn:aws:s3:::data-
	bucket/*"
	},
	{
	"Effect":
	"Allow",
	"Principa
	l": {
	"AWS": [
	
	"arn:aws:iam::<con
	sumer account id>:role/
	<role name>",
	
	"arn:aws:iam::<dat
	a account id>:role/
	service-role/AWSGl
	ueServiceRole-data-
	bucket-crawler"
	]
	},
	"Action":
	"s3:ListBucket",
	"Resource":
	"arn:aws:s3:::data-
	bucket"
	}
	]	

Operazione	Description	Competenze richieste
	} La bucket policy concede le autorizzazioni al ruolo IAM nell'account consumer e al ruolo del servizio crawler AWS Glue nell'account dati.	

Operazione	Description	Competenze richieste
(Se richiesto) Concedi l'accesso alla chiave di crittografia dei dati.	Se il bucket S3 è crittografato da una chiave AWS KMS, `kms:Decrypt` concedi l'autorizzazione sulla chiave al ruolo IAM nell'account consumer e al ruolo del servizio crawler AWS Glue nell'account dati. Aggiorna la policy chiave con la seguente dichiarazione: `{ "Effect": "Allow", "Principal": { "AWS": [ "arn:aws:iam::<consumer account id>:role/<role name>", "arn:aws:iam::<data account id>:role/service-role/AWSGlueServiceRole-data-bucket-crawler" ] }, "Action": "kms:Decrypt", "Resource": "arn:aws:kms:<region>:<data account id>:key/<key id>" }`	Amministratore cloud

Operazione	Description	Competenze richieste
Concedi al crawler l'accesso ai dati.	Allega la seguente policy IAM al ruolo di servizio del crawler: `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": "s3:GetObject", "Resource": "arn:aws:s3:::data-bucket/*" }, { "Effect": "Allow", "Action": "s3:ListBucket", "Resource": "arn:aws:s3:::data-bucket" } ] }`	Amministratore cloud

Operazione	Description	Competenze richieste
(Se richiesto) Concedi al crawler l'accesso alla chiave di crittografia dei dati.	Se il bucket S3 è crittografato da una chiave AWS KMS, `kms:Decrypt` concedi l'autorizzazione sulla chiave per il ruolo di servizio del crawler allegando la seguente policy: `{ "Effect": "Allow", "Action": "kms:Decrypt", "Resource": "arn:aws:kms:<region>:<data account id>:key/<key id>" }`	Amministratore del cloud

Operazione	Description	Competenze richieste
Concedi al ruolo IAM nell'account consumer e al crawler l'accesso al catalogo dati.	1. Accedi alla Console di gestione AWS e apri la console AWS Glue. 2. Nel pannello di navigazione, in Data Catalog, scegli Impostazioni. 3. Nella sezione Autorizzazioni, aggiungi la seguente dichiarazione, quindi scegli Salva. `{ "Version" : "2012-10-17", "Statement" : [ { "Effect" : "Allow", "Principal" : { "AWS" : [ "arn:aws:iam::<consumer account id>:role/<role name>", "arn:aws:iam::<data account id>:role/service-role/AWSGlueServiceRole-data-bucket-crawler" ] }, "Action" : "glue:*", "Resource" : [`	Amministratore cloud

Operazione	Description	Competenze richieste
	"arn:aws:glue:<region>:<data account id>:catalog", "arn:aws:glue:<region>:<data account id>:database/*", "arn:aws:glue:<region>:<data account id>:table/*"] }] } Questa policy consente tutte le azioni di AWS Glue su tutti i database e le tabelle nell'account dati. Puoi personalizzare la policy per concedere solo le autorizzazioni necessarie ai responsabili dei consumatori. Ad esempio, è possibile fornire l'accesso in sola lettura a tabelle o viste specifiche in un database.	

Accedi ai dati dall'account del consumatore

Operazione	Description	Competenze richieste
Crea un riferimento denominato per il catalogo dati.	Per creare un riferimento denominato al catalogo di dati, usa CloudShelluna CLI	Amministratore del cloud

Operazione	Description	Competenze richieste
	AWS installata localmente per eseguire il seguente comando: `aws athena create-data-catalog --name <shared catalog name> --type GLUE --parameters catalog-id=<data account id>`	

Operazione	Description	Competenze richieste
Concedi al ruolo IAM nell'account consumer l'accesso ai dati.	Allega la seguente policy al ruolo IAM nell'account consumer per concedere al ruolo l'accesso ai dati tra account diversi: `{ "Version": "2012-10-17", "Statement": [{ "Effect": "Allow", "Action": "s3:GetObject", "Resource": "arn:aws:s3:::data-bucket/*" }, { "Effect": "Allow", "Action": "s3:ListBucket", "Resource": "arn:aws:s3:::data-bucket" }, { "Effect": "Allow", "Action": "glue:*", "Resource": ["arn:aws:glue:<region>:<data account id>:catalog",`	Amministratore cloud

Operazione	Description	Competenze richieste

	<pre>"arn:aws:glue:<region>:<data account id>:database/*",
	
	"arn:aws:glue:<region>:<data account id>:table/*"
	]
	}
	]
	}</pre>
	Successivamente, utilizza il seguente modello per specificare quali utenti possono accettare il ruolo IAM nella relativa policy di fiducia:
	
	<pre>{
	"Version": "2012-10-17",
	"Statement": [
	{
	"Effect": "Allow",
	"Principal": {
	"AWS": "arn:aws:iam::<consumer account id>:user/<IAM user>"
	},
	"Action": "sts:AssumeRole"
	}
	]
	}</pre>	

Operazione	Description	Competenze richieste
	Infine, concedi agli utenti i permessi per assumere il ruolo IAM associando la stessa policy al gruppo di utenti a cui appartengono.	
(Se richiesto) Concedi al ruolo IAM nell'account consumer l'accesso alla chiave di crittografia dei dati.	Se il bucket S3 è crittografato da una chiave AWS KMS, `kms:Decrypt` concedi l'autorizzazione sulla chiave al ruolo IAM nell'account consumer allegando la seguente policy: `{ "Effect": "Allow", "Action": "kms:Decrypt", "Resource": "arn:aws:kms:<region>:<data account id>:key/<key id>" }`	Amministratore del cloud
Passa al ruolo IAM nell'account consumer per accedere ai dati.	In qualità di consumatore di dati, passa al ruolo IAM per accedere ai dati nell'account dati.	Consumatore di dati

Operazione	Description	Competenze richieste

Accedi ai dati.	Interroga i dati usando Athena. Ad esempio, apri l'editor di query Athena ed esegui la seguente query:
	`SELECT * FROM <shared catalog name>.<database name>.<table name>`
	Invece di utilizzare un riferimento denominato al catalogo, puoi fare riferimento al catalogo anche tramite il relativo Amazon Resource Name (ARN).
	Note
	Se utilizzi un riferimento dinamico al catalogo in una query o in una vista, racchiudi il riferimento tra virgolette doppie con escape (\»). Esempio:
	`SELECT * FROM \"glue:arn:aws:glue:<region>:<data account id>:catalog\".<database name>.<table name>`
		Consumatore di dati

Operazione	Description	Competenze richieste
	Per ulteriori informazioni, consulta Accesso da più account ai cataloghi di dati di AWS Glue nella Guida per l'utente di Amazon Athena.	

Risorse correlate

- Accesso da più account ai cataloghi di dati di AWS Glue (documentazione Athena)
- (AWS CLI) (riferimento ai comandi dell'interfaccia a riga di create-data-catalog comando AWS)
- Accesso a AWS Glue Data Catalog su più account con Amazon Athena (AWS Big Data Blog)
- Best practice di sicurezza in IAM (documentazione IAM)

Informazioni aggiuntive

Utilizzo di Lake Formation come alternativa per la condivisione tra account

Puoi anche utilizzare AWS Lake Formation per condividere l'accesso agli oggetti del catalogo AWS Glue tra gli account. Lake Formation offre un controllo granulare degli accessi a livello di colonna e riga, controllo degli accessi basato su tag, tabelle governate per le transazioni ACID e altre funzionalità. Sebbene Lake Formation sia ben integrato con Athena, richiede una configurazione aggiuntiva rispetto all'approccio basato esclusivamente sull'IAM di questo modello. Ti consigliamo di prendere in considerazione la decisione di utilizzare i controlli di accesso solo per Lake Formation o IAM nel contesto più ampio dell'architettura complessiva della tua soluzione. Le considerazioni includono quali altri servizi sono coinvolti e come si integrano con entrambi gli approcci.

Convertire la funzionalità temporale Teradata NORMALIZE in Amazon Redshift SQL

Po Hong, Amazon Web Services

Riepilogo

NORMALIZE è un'estensione Teradata dello standard ANSI SQL. Quando una tabella SQL include una colonna con un tipo di dati PERIOD, NORMALIZE combina i valori che corrispondono o si sovrappongono in quella colonna, per formare un unico periodo che consolida più valori di periodo individuali. Per utilizzare NORMALIZE, almeno una colonna nell'elenco SQL SELECT deve essere del tipo di dati TEMPORAL PERIOD di Teradata. Per ulteriori informazioni su NORMALIZE, vedere la documentazione di Teradata.

Amazon Redshift non supporta NORMALIZE, ma puoi implementare questa funzionalità utilizzando la sintassi SQL nativa e la funzione finestra LAG in Amazon Redshift. Questo modello si concentra sull'utilizzo dell'estensione Teradata NORMALIZE con la condizione ON MEETS OR OVERLAPS, che è il formato più popolare. Spiega come funziona questa funzionalità in Teradata e come può essere convertita nella sintassi SQL nativa di Amazon Redshift.

Prerequisiti e limitazioni

Prerequisiti

- Conoscenza ed esperienza di base di Teradata SQL
- Conoscenza ed esperienza in Amazon Redshift

Architecture

Stack tecnologico di origine

- Data warehouse Teradata

Stack tecnologico Target

- Amazon Redshift

Architettura di destinazione

Per un'architettura di alto livello per la migrazione di un database Teradata ad Amazon Redshift, consulta lo schema Migrare un database Teradata su Amazon Redshift utilizzando gli agenti di estrazione dati AWS SCT. La migrazione non converte automaticamente la frase Teradata NORMALIZE in Amazon Redshift SQL. Puoi convertire questa estensione Teradata seguendo le linee guida riportate in questo schema.

Tools (Strumenti)

Codice

Per illustrare il concetto e la funzionalità di NORMALIZE, si consideri la seguente definizione di tabella in Teradata:

```
CREATE TABLE systest.project
     (    emp_id        INTEGER,
          project_name  VARCHAR(20),
          dept_id       INTEGER,
          duration      PERIOD(DATE)
     );
```

Eseguite il seguente codice SQL per inserire dati di esempio nella tabella:

```
BEGIN TRANSACTION;

INSERT INTO systest.project VALUES (10, 'First Phase', 1000,  PERIOD(DATE '2010-01-10',
 DATE '2010-03-20') );
INSERT INTO systest.project VALUES (10, 'First Phase', 2000,  PERIOD(DATE '2010-03-20',
 DATE '2010-07-15') );

INSERT INTO systest.project VALUES (10, 'Second Phase', 2000,  PERIOD(DATE
 '2010-06-15', DATE '2010-08-18') );
INSERT INTO systest.project VALUES (20, 'First Phase', 2000,  PERIOD(DATE '2010-03-10',
 DATE '2010-07-20') );

INSERT INTO systest.project VALUES (20, 'Second Phase', 1000,  PERIOD(DATE
 '2020-05-10', DATE '2020-09-20') );

END TRANSACTION;
```

Risultati:

```
select * from systest.project order by 1,2,3;
```

```
 *** Query completed. 4 rows found. 4 columns returned.
 *** Total elapsed time was 1 second.

     emp_id  project_name              dept_id  duration
-----------  --------------------  -----------  ------------------------
         10  First Phase                  1000  ('10/01/10', '10/03/20')
         10  First Phase                  2000  ('10/03/20', '10/07/15')
         10  Second Phase                 2000  ('10/06/15', '10/08/18')
         20  First Phase                  2000  ('10/03/10', '10/07/20')
         20  Second Phase                 1000  ('20/05/10', '20/09/20')
```

Caso d'uso Teradata NORMALIZE

Ora aggiungi la clausola Teradata NORMALIZE SQL all'istruzione SELECT:

```
SELECT NORMALIZE ON MEETS OR OVERLAPS emp_id, duration
FROM systest.project
ORDER BY 1,2;
```

Questa operazione NORMALIZE viene eseguita su una singola colonna (). `emp_id` Infatti`emp_id=10`, i tre valori di periodo sovrapposti in termini di durata si fondono in un unico valore di periodo, come segue:

```
     emp_id  duration
-----------  ------------------------
         10  ('10/01/10', '10/08/18')
         20  ('10/03/10', '10/07/20')
         20  ('20/05/10', '20/09/20')
```

La seguente istruzione SELECT esegue un'operazione NORMALIZE su e. `project_name` `dept_id` Si noti che l'elenco SELECT contiene solo una colonna PERIOD,`duration`.

```
SELECT NORMALIZE project_name, dept_id, duration
FROM systest.project;
```

Output:

```
project_name              dept_id  duration
--------------------  -----------  ------------------------
```

```
First Phase                  1000  ('10/01/10', '10/03/20')
Second Phase                 1000  ('20/05/10', '20/09/20')
First Phase                  2000  ('10/03/10', '10/07/20')
Second Phase                 2000  ('10/06/15', '10/08/18')
```

SQL equivalente ad Amazon Redshift

Amazon Redshift attualmente non supporta il tipo di dati PERIOD in una tabella. È invece necessario dividere un campo di dati TERADATA PERIOD in due parti:`start_date, end_date`, come segue:

```
CREATE TABLE systest.project
     (    emp_id        INTEGER,
          project_name  VARCHAR(20),
          dept_id       INTEGER,
          start_date  DATE,
          end_date    DATE
     );
```

Inserire dati di esempio nella tabella:

```
BEGIN TRANSACTION;

INSERT INTO systest.project VALUES (10, 'First Phase', 1000,  DATE '2010-01-10', DATE
 '2010-03-20' );
INSERT INTO systest.project VALUES (10, 'First Phase', 2000,  DATE '2010-03-20', DATE
 '2010-07-15');

INSERT INTO systest.project VALUES (10, 'Second Phase', 2000,  DATE '2010-06-15', DATE
 '2010-08-18' );
INSERT INTO systest.project VALUES (20, 'First Phase', 2000,  DATE '2010-03-10', DATE
 '2010-07-20' );

INSERT INTO systest.project VALUES (20, 'Second Phase', 1000,  DATE '2020-05-10', DATE
 '2020-09-20' );

END TRANSACTION;
```

Output:

```
 emp_id | project_name | dept_id | start_date |  end_date
--------+--------------+---------+------------+------------
```

```
     10 | First Phase  |    1000 | 2010-01-10 | 2010-03-20
     10 | First Phase  |    2000 | 2010-03-20 | 2010-07-15
     10 | Second Phase |    2000 | 2010-06-15 | 2010-08-18
     20 | First Phase  |    2000 | 2010-03-10 | 2010-07-20
     20 | Second Phase |    1000 | 2020-05-10 | 2020-09-20
(5 rows)
```

Per riscrivere la clausola NORMALIZE di Teradata, puoi utilizzare la funzione LAG window in Amazon Redshift. Questa funzione restituisce i valori di una riga con un determinato offset al di sopra (prima) della riga corrente nella partizione.

È possibile utilizzare la funzione LAG per identificare ogni riga che inizia un nuovo periodo determinando se un periodo corrisponde o si sovrappone al periodo precedente (0 se sì e 1 se no). Quando questo flag viene sommato cumulativamente, fornisce un identificatore di gruppo che può essere utilizzato nella clausola Group By esterna per ottenere il risultato desiderato in Amazon Redshift.

Ecco un esempio di istruzione SQL di Amazon Redshift che utilizza LAG ():

```
SELECT emp_id, start_date, end_date,
            (CASE WHEN start_date <= LAG(end_date) OVER (PARTITION BY emp_id ORDER BY
 start_date, end_date) THEN 0 ELSE 1 END) AS GroupStartFlag
FROM systest.project
ORDER BY 1,2;
```

Output:

```
 emp_id | start_date |  end_date  | groupstartflag
--------+------------+------------+----------------
     10 | 2010-01-10 | 2010-03-20 |              1
     10 | 2010-03-20 | 2010-07-15 |              0
     10 | 2010-06-15 | 2010-08-18 |              0
     20 | 2010-03-10 | 2010-07-20 |              1
     20 | 2020-05-10 | 2020-09-20 |              1
(5 rows)
```

La seguente istruzione SQL di Amazon Redshift si normalizza solo sulla colonna: `emp_id`

```
SELECT T2.emp_id, MIN(T2.start_date) as new_start_date, MAX(T2.end_date) as
 new_end_date
FROM
```

```
( SELECT T1.*, SUM(GroupStartFlag) OVER (PARTITION BY emp_id ORDER BY start_date ROWS
 UNBOUNDED PRECEDING) As GroupID
FROM ( SELECT emp_id, start_date, end_date,
            (CASE WHEN start_date <= LAG(end_date) OVER (PARTITION BY emp_id ORDER BY
 start_date, end_date) THEN 0 ELSE 1 END) AS GroupStartFlag
FROM systest.project ) T1
) T2
GROUP BY T2.emp_id, T2.GroupID
ORDER BY 1,2;
```

Output:

```
 emp_id | new_start_date | new_end_date
--------+----------------+----------------------------------
     10 | 2010-01-10     | 2010-08-18
     20 | 2010-03-10     | 2010-07-20
     20 | 2020-05-10     | 2020-09-20
(3 rows)
```

La seguente istruzione SQL di Amazon Redshift si normalizza su entrambe le `project_name` colonne e: `dept_id`

```
SELECT T2.project_name, T2.dept_id, MIN(T2.start_date) as new_start_date,
 MAX(T2.end_date) as new_end_date
FROM
( SELECT T1.*, SUM(GroupStartFlag) OVER (PARTITION BY project_name, dept_id ORDER BY
 start_date ROWS UNBOUNDED PRECEDING) As GroupID
FROM ( SELECT project_name, dept_id, start_date, end_date,
            (CASE WHEN start_date <= LAG(end_date) OVER (PARTITION BY project_name,
 dept_id ORDER BY start_date, end_date) THEN 0 ELSE 1 END) AS GroupStartFlag
FROM systest.project ) T1
) T2
GROUP BY T2.project_name, T2.dept_id, T2.GroupID
ORDER BY 1,2,3;
```

Output:

```
 project_name | dept_id | new_start_date | new_end_date
--------------+---------+----------------+--------------
 First Phase  |    1000 | 2010-01-10     | 2010-03-20
```

```
 First Phase  |    2000 | 2010-03-10     | 2010-07-20
 Second Phase |    1000 | 2020-05-10     | 2020-09-20
 Second Phase |    2000 | 2010-06-15     | 2010-08-18
(4 rows)
```

Epiche

Convertire NORMALIZE in Amazon Redshift SQL

Operazione	Description	Competenze richieste
Crea il tuo codice Teradata SQL.	Usa la frase NORMALIZE in base alle tue esigenze.	SQL Developer
Converti il codice in Amazon Redshift SQL.	Per convertire il codice, segui le linee guida nella sezione «Strumenti» di questo modello.	SQL Developer
Esegui il codice in Amazon Redshift.	Crea la tua tabella, carica i dati nella tabella ed esegui il codice in Amazon Redshift.	SQL Developer

Risorse correlate

Riferimenti

- Funzione temporale Teradata NORMALIZE (documentazione Teradata)
- Funzione finestra LAG (documentazione Amazon Redshift)
- Esegui la migrazione ad Amazon Redshift (sito web AWS)
- Esegui la migrazione di un database Teradata su Amazon Redshift utilizzando agenti di estrazione dati AWS SCT (AWS Prescriptive Guidance)
- Conversione della funzionalità Teradata RESET WHEN in Amazon Redshift SQL (AWS Prescriptive Guidance)

Strumenti

- Strumento di conversione dello schema AWS (AWS SCT)

Partner

- Partner AWS con competenze per la migrazione

Convertire la funzionalità Teradata RESET WHEN in Amazon Redshift SQL

Po Hong, Amazon Web Services

Riepilogo

RESET WHEN è una funzionalità di Teradata utilizzata nelle funzioni analitiche delle finestre SQL. È un'estensione dello standard ANSI SQL. RESET WHEN determina la partizione su cui opera una funzione di finestra SQL in base a una condizione specificata. Se la condizione restituisce TRUE, viene creata una nuova sottopartizione dinamica all'interno della partizione di finestra esistente. Per ulteriori informazioni su RESET WHEN, consultate la documentazione di Teradata.

Amazon Redshift non supporta RESET WHEN nelle funzioni delle finestre SQL. Per implementare questa funzionalità, devi convertire RESET WHEN nella sintassi SQL nativa in Amazon Redshift e utilizzare più funzioni annidate. Questo modello dimostra come utilizzare la funzionalità Teradata RESET WHEN e come convertirla nella sintassi SQL di Amazon Redshift.

Prerequisiti e limitazioni

Prerequisiti

- Conoscenza di base del data warehouse Teradata e della sua sintassi SQL
- Buona conoscenza di Amazon Redshift e della sua sintassi SQL

Architecture

Stack tecnologico di origine

- Data warehouse Teradata

Stack tecnologico Target

- Amazon Redshift

Architettura

Per un'architettura di alto livello per la migrazione di un database Teradata ad Amazon Redshift, consulta lo schema Migrare un database Teradata su Amazon Redshift utilizzando gli agenti di estrazione dati AWS SCT. La migrazione non converte automaticamente la frase Teradata RESET WHEN in Amazon Redshift SQL. Puoi convertire questa estensione Teradata seguendo le linee guida nella sezione successiva.

Tools (Strumenti)

Codice

Per illustrare il concetto di RESET WHEN, si consideri la seguente definizione di tabella in Teradata:

```
create table systest.f_account_balance
( account_id integer NOT NULL,
  month_id integer,
  balance integer )
unique primary index (account_id, month_id);
```

Esegui il seguente codice SQL per inserire dati di esempio nella tabella:

```
BEGIN TRANSACTION;
Insert Into systest.f_account_balance values (1,1,60);
Insert Into systest.f_account_balance values (1,2,99);
Insert Into systest.f_account_balance values (1,3,94);
Insert Into systest.f_account_balance values (1,4,90);
Insert Into systest.f_account_balance values (1,5,80);
Insert Into systest.f_account_balance values (1,6,88);
Insert Into systest.f_account_balance values (1,7,90);
Insert Into systest.f_account_balance values (1,8,92);
Insert Into systest.f_account_balance values (1,9,10);
Insert Into systest.f_account_balance values (1,10,60);
Insert Into systest.f_account_balance values (1,11,80);
Insert Into systest.f_account_balance values (1,12,10);
END TRANSACTION;
```

La tabella di esempio contiene i seguenti dati:

account_id	month_id	balance
1	1	60

1	2	99
1	3	94
1	4	90
1	5	80
1	6	88
1	7	90
1	8	92
1	9	10
1	10	60
1	11	80
1	12	10

Per ogni account, supponiamo che tu voglia analizzare la sequenza di aumenti mensili consecutivi del saldo. Quando il saldo di un mese è inferiore o uguale al saldo del mese precedente, è necessario azzerare il contatore e riavviare il sistema.

Caso d'uso Teradata RESET WHEN

Per analizzare questi dati, Teradata SQL utilizza una funzione finestra con un aggregato annidato e una frase RESET WHEN, come segue:

```
SELECT account_id, month_id, balance,
 ( ROW_NUMBER() OVER (PARTITION BY account_id ORDER BY month_id
RESET WHEN balance <= SUM(balance) over (PARTITION BY account_id ORDER BY month_id ROWS
 BETWEEN 1 PRECEDING AND 1 PRECEDING) ) -1 ) as balance_increase
FROM systest.f_account_balance
ORDER BY 1,2;
```

Output:

account_id	id_mese	balance	balance_increase
1	1	60	0
1	2	99	1
1	3	94	0
1	4	90	0
1	5	80	0
1	6	88	1
1	7	90	2
1	8	92	3
1	9	10	0
1	10	60	1
1	11	80	2
1	12	10	0

La query viene elaborata come segue in Teradata:

1. La funzione di aggregazione SUM (saldo) calcola la somma di tutti i saldi di un determinato conto in un determinato mese.
2. Controlliamo se un saldo in un determinato mese (per un determinato account) è maggiore del saldo del mese precedente.
3. Se il saldo aumenta, tracciamo un valore di conteggio cumulativo. Se la condizione RESET WHEN risulta falsa, il che significa che il saldo è aumentato nei mesi successivi, continuiamo ad aumentare il conteggio.
4. La funzione analitica ordinata ROW_NUMBER () calcola il valore del conteggio. Quando raggiungiamo un mese il cui saldo è inferiore o uguale al saldo del mese precedente, la condizione RESET WHEN risulta vera. In tal caso, iniziamo una nuova partizione e ROW_NUMBER ()

riavvia il conteggio da 1. Utilizziamo ROWS BETWEEN 1 PRECEDING AND 1 PRECEDING per accedere al valore della riga precedente.

5. Sottraiamo 1 per assicurarci che il valore del conteggio inizi con 0.

SQL equivalente ad Amazon Redshift

Amazon Redshift non supporta la frase RESET WHEN in una funzione di finestra analitica SQL. Per ottenere lo stesso risultato, è necessario riscrivere Teradata SQL utilizzando la sintassi SQL nativa di Amazon Redshift e sottoquery annidate, come segue:

```
SELECT account_id, month_id, balance,
   (ROW_NUMBER() OVER(PARTITION BY account_id, new_dynamic_part ORDER BY month_id) -1)
 as balance_increase
FROM
( SELECT account_id, month_id, balance, prev_balance,
SUM(dynamic_part) OVER (PARTITION BY account_id ORDER BY month_id ROWS BETWEEN
 UNBOUNDED PRECEDING AND CURRENT ROW) As new_dynamic_part
FROM ( SELECT account_id, month_id, balance,
SUM(balance) over (PARTITION BY account_id ORDER BY month_id ROWS BETWEEN 1 PRECEDING
 AND 1 PRECEDING) as prev_balance,
(CASE When balance <= prev_balance Then 1 Else 0 END) as dynamic_part
FROM systest.f_account_balance ) A
) B
ORDER BY 1,2;
```

Poiché Amazon Redshift non supporta le funzioni di finestra annidata nella clausola SELECT di una singola istruzione SQL, è necessario utilizzare due sottoquery annidate.

- Nella sottoquery interna (alias A), viene creato e popolato un indicatore di partizione dinamica (dynamic_part). dynamic_part è impostato su 1 se il saldo di un mese è inferiore o uguale al saldo del mese precedente; in caso contrario, è impostato su 0.
- Nel livello successivo (alias B), viene generato un attributo new_dynamic_part come risultato di una funzione della finestra SUM.
- Infine, aggiungete new_dynamic_part come nuovo attributo di partizione (partizione dinamica) all'attributo di partizione esistente (account_id) e applicate la stessa funzione di finestra ROW_NUMBER () di Teradata (e meno una).

Dopo queste modifiche, Amazon Redshift SQL genera lo stesso output di Teradata.

Epiche

Converti RESET WHEN in Amazon Redshift SQL

Operazione	Description	Competenze richieste
Crea la tua funzione di finestra Teradata.	Usa gli aggregati annidati e la frase RESET WHEN in base alle tue esigenze.	SQL Developer
Converti il codice in Amazon Redshift SQL.	Per convertire il codice, segui le linee guida nella sezione «Strumenti» di questo modello.	SQL Developer
Esegui il codice in Amazon Redshift.	Crea la tua tabella, carica i dati nella tabella ed esegui il codice in Amazon Redshift.	SQL Developer

Risorse correlate

Riferimenti

- RESET WHEN Phrase (documentazione Teradata)
- Spiegazione RESET WHEN (Stack Overflow)
- Esegui la migrazione ad Amazon Redshift (sito web AWS)
- Esegui la migrazione di un database Teradata su Amazon Redshift utilizzando agenti di estrazione dati AWS SCT (AWS Prescriptive Guidance)
- Conversione della funzionalità temporale Teradata NORMALIZE in Amazon Redshift SQL (AWS Prescriptive Guidance)

Strumenti

- Strumento di conversione dello schema AWS (AWS SCT)

Partner

- Partner AWS con competenze per la migrazione

Implementa e gestisci un data lake serverless sul cloud AWS utilizzando l'infrastruttura come codice

Kirankumar Chandrashekar e Abdel Jaidi, Amazon Web Services

Riepilogo

Questo modello descrive come utilizzare l'elaborazione e l'infrastruttura senza server come codice (IaC) per implementare e amministrare un data lake sul cloud Amazon Web Services (AWS). Questo modello si basa sul workshop Serverless Data Lake Framework (SDLF) sviluppato da AWS.

SDLF è una raccolta di risorse riutilizzabili che accelera la distribuzione di data lake aziendali sul cloud AWS e aiuta a velocizzare la distribuzione alla produzione. Viene utilizzato per implementare la struttura di base di un data lake seguendo le migliori pratiche.

SDLF implementa un processo di integrazione continua/distribuzione continua (CI/CD) in tutta la distribuzione del codice e dell'infrastruttura utilizzando servizi AWS come AWS, AWS e CodePipeline AWS. CodeBuild CodeCommit

Questo modello utilizza più servizi serverless AWS per semplificare la gestione dei data lake. Questi includono Amazon Simple Storage Service (Amazon S3) e Amazon DynamoDB per lo storage, AWS Lambda e AWS Glue per l'informatica e Amazon Events, Amazon Simple Queue Service (Amazon SQS) CloudWatch e AWS Step Functions per l'orchestrazione.

I servizi di codice AWS CloudFormation e AWS fungono da livello IaC per fornire distribuzioni riproducibili e veloci con operazioni e amministrazione semplici.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- AWS Command Line Interface (AWS CLI), installata e configurata.
- Un client Git, installato e configurato.
- Il workshop SDLF, aperto in una finestra del browser Web e pronto per l'uso.

Architecture

Il diagramma di architettura illustra un processo basato sugli eventi con i seguenti passaggi.

1. Dopo aver aggiunto un file al bucket S3 di dati grezzi, una notifica di evento Amazon S3 viene inserita in una coda SQS. Ogni notifica viene consegnata come file JSON, che contiene metadati come il nome del bucket S3, la chiave dell'oggetto o il timestamp.
2. Questa notifica viene utilizzata da una funzione Lambda che indirizza l'evento al processo di estrazione, trasformazione e caricamento (ETL) corretto in base ai metadati. La funzione Lambda può anche utilizzare configurazioni contestuali archiviate in una tabella Amazon DynamoDB. Questo passaggio consente il disaccoppiamento e la scalabilità su più applicazioni nel data lake.
3. L'evento viene indirizzato alla prima funzione Lambda del processo ETL, che trasforma e sposta i dati dall'area dei dati grezzi all'area di staging per il data lake. Il primo passo consiste nell'aggiornare il catalogo completo. Questa è una tabella DynamoDB che contiene tutti i metadati dei file del data lake. Ogni riga di questa tabella contiene metadati operativi su un singolo oggetto archiviato in Amazon S3. Viene effettuata una chiamata sincrona a una funzione Lambda che esegue una trasformazione della luce, un'operazione computazionalmente poco costosa (come la conversione di un file da un formato all'altro), sull'oggetto S3. Poiché è stato aggiunto un nuovo oggetto al bucket S3 di staging, il catalogo completo viene aggiornato e viene inviato un messaggio alla coda SQS in attesa della fase successiva dell'ETL.
4. Una regola CloudWatch Events attiva una funzione Lambda ogni 5 minuti. Questa funzione verifica se i messaggi sono stati recapitati alla coda SQS dalla fase ETL precedente. Se è stato recapitato un messaggio, la funzione Lambda avvia la seconda funzione di AWS Step Functions nel processo ETL.
5. Una trasformazione pesante viene quindi applicata a un batch di file. Questa trasformazione complessa è un'operazione computazionalmente costosa, come una chiamata sincrona a un job AWS Glue, un'attività AWS Fargate, una fase Amazon EMR o un notebook Amazon. SageMaker I metadati delle tabelle vengono estratti dai file di output utilizzando un crawler AWS Glue, che aggiorna il catalogo AWS Glue. I metadati dei file vengono inoltre aggiunti alla tabella di catalogo completa in DynamoDB. Infine, viene eseguita anche una fase di qualità dei dati che sfrutta Deequ.

Stack tecnologico

- CloudWatch Eventi Amazon
- AWS CloudFormation
- AWS CodePipeline

- AWS CodeBuild
- AWS CodeCommit
- Amazon DynamoDB
- AWS Glue
- AWS Lambda
- Simple Storage Service (Amazon S3)
- Amazon SQS
- AWS Step Functions

Tools (Strumenti)

- Amazon CloudWatch Events — CloudWatch Events offre un flusso quasi in tempo reale di eventi di sistema che descrivono i cambiamenti nelle risorse AWS.
- AWS CloudFormation: CloudFormation aiuta a creare e fornire implementazioni di infrastrutture AWS in modo prevedibile e ripetuto.
- AWS CodeBuild: CodeBuild è un servizio di build completamente gestito che compila il codice sorgente, esegue test unitari e produce artefatti pronti per la distribuzione.
- AWS CodeCommit: CodeCommit è un servizio di controllo delle versioni ospitato da AWS che puoi utilizzare per archiviare e gestire risorse private (come codice sorgente e file binari).
- AWS CodePipeline: CodePipeline è un servizio di distribuzione continua che puoi utilizzare per modellare, visualizzare e automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- Amazon DynamoDB — DynamoDB è un servizio di database NoSQL completamente gestito che offre prestazioni veloci e prevedibili con scalabilità.
- AWS Glue — AWS Glue è un servizio ETL completamente gestito che semplifica la preparazione e il caricamento dei dati per l'analisi.
- AWS Lambda — Lambda supporta l'esecuzione di codice senza effettuare il provisioning o la gestione di server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti altamente scalabile. Amazon S3 può essere utilizzato per un'ampia gamma di soluzioni di storage, tra cui siti Web, applicazioni mobili, backup e data lake.

- AWS Step Functions - AWS Step Functions è un orchestratore di funzioni senza server che semplifica la sequenza delle funzioni AWS Lambda e di più servizi AWS in applicazioni aziendali critiche.
- Amazon SQS — Amazon Simple Queue Service (Amazon SQS) è un servizio di accodamento dei messaggi completamente gestito che ti aiuta a disaccoppiare e scalare microservizi, sistemi distribuiti e applicazioni serverless.
- Deequ — Deequ è uno strumento che ti aiuta a calcolare i parametri di qualità dei dati per set di dati di grandi dimensioni, definire e verificare i vincoli di qualità dei dati e rimanere informato sui cambiamenti nella distribuzione dei dati.

Archivio di codice

Il codice sorgente e le risorse per SDLF sono disponibili nel repository AWS Labs. GitHub

Epiche

Configura la CI/CD pipeline per il provisioning di IAc

Operazione	Description	Competenze richieste
Configura la CI/CD pipeline per gestire IAc per il data lake.	Accedi alla Console di gestione AWS e segui i passaggi della sezione Configurazione iniziale del workshop SDLF. Questo crea le CI/CD risorse iniziali, come CodeCommit repository, CodeBuild ambienti e CodePipeline pipeline che forniscono e gestiscono IaC per il data lake.	DevOps ingegnere

Controllo della versione dell'IaC

Operazione	Description	Competenze richieste
Clona il CodeCommit repository sul tuo computer locale.	Seguite i passaggi indicati nella sezione Implement azione delle basi del workshop SDLF. Questo ti aiuta a clonare il repository Git che ospita IAc nel tuo ambiente locale. Per ulteriori informazioni, consulta Connessione ai CodeCommit repository dalla documentazione. CodeCommit	DevOps ingegnere
Modifica i CloudFormation modelli.	Usa la tua workstation locale e un editor di codice per modificare i CloudFormation modelli in base ai tuoi casi d'uso o ai tuoi requisiti. Inviali nel repository Git clonato localmente. Per ulteriori informazioni, consulta Working with AWS CloudFormation templates dalla CloudFormation documentazione AWS.	DevOps ingegnere
Invia le modifiche al CodeCommit repository.	Il codice dell'infrastruttura è ora sotto il controllo della versione e le modifiche alla base di codice vengono tracciate. Quando invii una modifica al CodeCommi	DevOps ingegnere

Operazione	Description	Competenze richieste
	t repository, la applica CodePipeline automatic amente all'infrastruttura e la invia a. CodeBuild **Important** Se utilizzi la CLI AWS SAM in CodeBuild, esegui i comandi `sam package` and`sam deploy`. Se usi l'interfaccia a riga di comando di AWS, esegui i comandi `aws cloudformation package` and`aws cloudformation deploy`.	

Risorse correlate

Configura la CI/CD pipeline per il provisioning di IaC

- Workshop SDLF — Configurazione iniziale

Controllo della versione dell'IAc

- Workshop SDLF — Implementazione delle basi
- Connessione ai repository CodeCommit
- Lavorare con i CloudFormation modelli AWS

Altre risorse

- Architettura di riferimento della pipeline di analisi dei dati senza server AWS
- Documentazione SDLF

Applica l'etichettatura dei cluster Amazon EMR al momento del lancio

Priyanka Chaudhary, Amazon Web Services

Riepilogo

Questo modello fornisce un controllo di sicurezza che garantisce che i cluster Amazon EMR siano etichettati al momento della creazione.

Amazon EMR è un servizio Amazon Web Services (AWS) per l'elaborazione e l'analisi di grandi quantità di dati. Amazon EMR offre un servizio espandibile e a bassa configurazione come alternativa più semplice all'esecuzione interna del cluster computing. Puoi utilizzare i tag per classificare le risorse AWS in diversi modi, ad esempio per scopo, proprietario o ambiente. Ad esempio, puoi etichettare i tuoi cluster Amazon EMR assegnando metadati personalizzati a ciascun cluster. Un tag è composto da una chiave e da un valore definiti dall'utente. Ti consigliamo di creare un set coerente di tag per soddisfare i requisiti della tua organizzazione. Quando aggiungi un tag a un cluster Amazon EMR, il tag viene anche propagato a ogni istanza attiva di Amazon Elastic Compute Cloud (Amazon EC2) associata al cluster. Allo stesso modo, quando rimuovi un tag da un cluster Amazon EMR, tale tag viene rimosso anche da ogni EC2 istanza attiva associata.

Il controllo investigativo monitora le chiamate API e avvia un evento Amazon CloudWatch Events per RunJobFlow, AddTagsRemoveTags, e. CreateTags APIs L'evento chiama AWS Lambda, che esegue uno script Python. La funzione Python ottiene l'ID del cluster Amazon EMR dall'input JSON dell'evento ed esegue i seguenti controlli:

- Verifica se il cluster Amazon EMR è configurato con i nomi di tag che hai specificato.
- In caso contrario, invia una notifica Amazon Simple Notification Service (Amazon SNS) all'utente con le informazioni pertinenti: il nome del cluster Amazon EMR, i dettagli della violazione, la regione AWS, l'account AWS e Amazon Resource Name (ARN) per Lambda da cui proviene questa notifica.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo

- Un bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) per caricare il codice Lambda fornito. In alternativa, puoi creare un bucket S3 per questo scopo, come descritto nella sezione Epics.
- Un indirizzo email attivo a cui desideri ricevere notifiche di violazione.
- Un elenco di tag obbligatori che desideri controllare.

Limitazioni

- Questo controllo di sicurezza è regionale. Devi distribuirlo in ogni regione AWS che desideri monitorare.

Versioni del prodotto

- Amazon EMR versione 4.8.0 e successive.

Architecture

Architettura del workflow

Automazione e scalabilità

- Se utilizzi AWS Organizations, puoi utilizzare AWS Cloudformation StackSets per distribuire questo modello in più account che desideri monitorare.

Tools (Strumenti)

Servizi AWS

- AWS CloudFormation: AWS ti CloudFormation aiuta a modellare e configurare le tue risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle per tutto il loro ciclo di vita. Puoi utilizzare un modello per descrivere le tue risorse e le loro dipendenze e lanciarle e configurarle insieme come uno stack, invece di gestire le risorse singolarmente. Puoi gestire e fornire stack su più account AWS e regioni AWS.
- Amazon CloudWatch Events - Amazon CloudWatch Events offre un flusso quasi in tempo reale di eventi di sistema che descrivono i cambiamenti nelle risorse AWS.

- Amazon EMR - Amazon EMR è un servizio web che semplifica l'esecuzione di framework di big data e l'elaborazione di grandi quantità di dati in modo efficiente.
- AWS Lambda: AWS Lambda è un servizio di elaborazione che supporta l'esecuzione di codice senza effettuare il provisioning o la gestione di server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti. È possibile utilizzare Amazon S3 per memorizzare e recuperare qualsiasi volume di dati, in qualunque momento e da qualunque luogo tramite il Web.
- Amazon SNS — Amazon Simple Notification Service (Amazon SNS) coordina e gestisce la consegna o l'invio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail. I sottoscrittori ricevono tutti gli stessi messaggi pubblicati sugli argomenti ai quali sono hanno effettuato la sottoscrizione.

Codice

Questo modello include i seguenti allegati:

- `EMRTagValidation.zip`— Il codice Lambda per il controllo di sicurezza.
- `EMRTagValidation.yml`— Il CloudFormation modello che configura l'evento e la funzione Lambda.

Epiche

Configura il bucket S3

Operazione	Description	Competenze richieste
Definisci il bucket S3.	Sulla console Amazon S3, scegli o crea un bucket S3 per ospitare il file.zip con codice Lambda. Questo bucket S3 deve trovarsi nella stessa regione AWS del cluster Amazon EMR che desideri monitorare. Il nome di un bucket Amazon S3 è univoco	Architetto del cloud

Operazione	Description	Competenze richieste
	a livello globale e lo spazio dei nomi è condiviso da tutti gli account AWS. Il nome del bucket S3 non può includere barre iniziali.	
Carica il codice Lambda.	Carica il file.zip con codice Lambda fornito nella sezione Allegati nel bucket S3.	Architetto del cloud

Implementa il modello AWS CloudFormation

Operazione	Description	Competenze richieste
Avvia il CloudFormation modello AWS.	Apri la CloudFormation console AWS nella stessa regione AWS del bucket S3 e distribuisci il modello. Per ulteriori informazioni sulla distribuzione di CloudFormation modelli AWS, consulta Creazione di uno stack sulla CloudFormation console AWS nella CloudFormation documentazione.	Architetto del cloud
Completa i parametri nel modello.	Quando avvii il modello, ti verranno richieste le seguenti informazioni: • Bucket S3: specifica il bucket che hai creato o selezionato nella prima epic. Qui è dove hai caricato il	Architetto del cloud

Operazione	Description	Competenze richieste
	codice Lambda allegato (file.zip). • Chiave S3: specifica la posizione del file Lambda .zip nel bucket S3 (ad esempio, filename .zip o controls/ filename .zip). Non includere le barre iniziali. • E-mail di notifica: fornisci un indirizzo e-mail attivo a cui desideri ricevere le notifiche di Amazon SNS. • Etichettatura dei nomi delle chiavi: fornisci i tag che desideri controllare in un elenco separato da virgole (ad esempio,,). `ApplicationID Environment Owner` L'evento CloudWatch Events monitora il cluster alla ricerca di questi tag e invia una notifica se non vengono trovati. • Livello di registrazione Lambda: specifica il livello e la frequenza di registrazione per la funzione Lambda. Utilizzate Info per registrare messaggi informativi dettagliati sullo stato di avanzamento, Errore per gli eventi di errore che potrebbero comunque consentire la continuazione	

Operazione	Description	Competenze richieste
	della distribuzione e Avviso per situazioni potenzial mente dannose.	

Confermare la sottoscrizione

Operazione	Description	Competenze richieste
Confermare la sottoscrizione.	Quando il CloudFormation modello viene distribuito correttamente, invia un'e-mail di iscrizione all'indirizzo e-mail fornito. È necessario confermare questa sottoscri zione e-mail per iniziare a ricevere notifiche di violazione.	Architetto del cloud

Risorse correlate

- Guida per sviluppatori AWS Lambda
- Etichettatura dei cluster in Amazon EMR

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Assicurati che la registrazione di Amazon EMR su Amazon S3 sia abilitata al momento del lancio

Priyanka Chaudhary, Amazon Web Services

Riepilogo

Questo modello fornisce un controllo di sicurezza che monitora la configurazione di registrazione per i cluster Amazon EMR in esecuzione su Amazon Web Services (AWS).

Amazon EMR è uno strumento AWS per l'elaborazione e l'analisi di big data. Amazon EMR offre il servizio espandibile a bassa configurazione come alternativa all'esecuzione interna del cluster computing. Amazon EMR offre due tipi di cluster EMR.

- Cluster Amazon EMR transitori: i cluster Amazon EMR transitori si spengono automaticamente e smettono di incorrere in costi al termine dell'elaborazione.
- Cluster Amazon EMR persistenti: i cluster Amazon EMR persistenti continuano a funzionare dopo il completamento del processo di elaborazione dei dati.

Amazon EMR e Hadoop producono entrambi file di log che comunicano lo stato sul cluster. Per impostazione predefinita, questi vengono scritti nel nodo master nella directory//. mnt/var/log A seconda di come configuri il cluster al momento dell'avvio, puoi anche salvare questi log su Amazon Simple Storage Service (Amazon S3) e visualizzarli tramite lo strumento grafico di debug. Tieni presente che la registrazione dei log di Amazon S3 può essere specificata solo all'avvio del cluster. Con questa configurazione, i log vengono inviati dal nodo primario alla posizione Amazon S3 ogni 5 minuti. Per i cluster transitori, la registrazione di Amazon S3 è importante perché i cluster scompaiono al termine dell'elaborazione e questi file di registro possono essere utilizzati per eseguire il debug di eventuali lavori non riusciti.

Il modello utilizza un CloudFormation modello AWS per implementare un controllo di sicurezza che monitora le chiamate API e avvia Amazon CloudWatch Events su "»RunJobFlow. Il trigger richiama AWS Lambda, che esegue uno script Python. La funzione Lambda recupera l'ID del cluster EMR dall'input JSON dell'evento e verifica anche la presenza di un URI di log di Amazon S3. Se non viene trovato un URI Amazon S3, la funzione Lambda invia una notifica Amazon Simple Notification Service (Amazon SNS) con i dettagli del nome del cluster EMR, i dettagli della violazione, la regione AWS, l'account AWS e il nome Lambda Amazon Resource Name (ARN) da cui proviene la notifica.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un bucket S3 per il file.zip del codice Lambda
- Un indirizzo e-mail a cui desideri ricevere la notifica di violazione

Limitazioni

- Questo controllo investigativo è regionale e deve essere distribuito nelle regioni AWS che intendi monitorare.

Versioni del prodotto

- Amazon EMR versione 4.8.0 e successive

Architecture

Stack tecnologico Target

- Evento Amazon CloudWatch Events
- Amazon EMR
- funzione Lambda
- Bucket S3
- Amazon SNS

Architettura Target

Automazione e scalabilità

- Se utilizzi AWS Organizations, puoi utilizzare AWS CloudFormation StackSets per distribuire questo modello in più account che desideri monitorare.

Tools (Strumenti)

Strumenti

- AWS CloudFormation: AWS ti CloudFormation aiuta a modellare e configurare le risorse AWS utilizzando l'infrastruttura come codice.
- AWS Cloudwatch Events: AWS CloudWatch Events offre un flusso quasi in tempo reale di eventi di sistema che descrivono i cambiamenti nelle risorse AWS.
- Amazon EMR: Amazon EMR è una piattaforma di cluster gestita che semplifica l'esecuzione di framework di big data.
- AWS Lambda: AWS Lambda supporta l'esecuzione di codice senza effettuare il provisioning o la gestione di server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo.
- Amazon S3 — Amazon S3 è un'interfaccia di servizi Web che puoi utilizzare per archiviare e recuperare qualsiasi quantità di dati da qualsiasi punto del Web.
- Amazon SNS: Amazon SNS è un servizio Web che coordina e gestisce la consegna o l'invio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.

Codice

- Un file.zip del progetto è disponibile come allegato.

Epiche

Definisci il bucket S3

Operazione	Description	Competenze richieste
Definisci il bucket S3.	Per ospitare il file.zip con codice Lambda, scegli o crea un bucket S3 con un nome univoco che non contenga barre iniziali. Il nome di un bucket S3 è unico a livello globale e lo spazio dei nomi è condiviso da tutti gli account	Architetto del cloud

Operazione	Description	Competenze richieste
	AWS. Il bucket S3 deve trovarsi nella stessa regione AWS del cluster Amazon EMR oggetto di valutazione.	

Carica il codice Lambda nel bucket S3

Operazione	Description	Competenze richieste
Carica il codice Lambda nel bucket S3.	Carica il file.zip con codice Lambda fornito nella sezione «Allegati» nel bucket S3. Il bucket S3 deve trovarsi nella stessa regione del cluster Amazon EMR in fase di valutazione.	Architetto del cloud

Implementa il modello AWS CloudFormation

Operazione	Description	Competenze richieste
Implementa il CloudFormation modello AWS.	Sulla CloudFormation console AWS, nella stessa regione del bucket S3, distribuisci il CloudFormation modello AWS fornito come allegato a questo pattern. Nella prossima epopea, fornisci i valori per i parametri. Per ulteriori informazioni sulla distribuzione di CloudFormation modelli AWS, consulta la sezione «Risorse correlate».	Architetto del cloud

Completa i parametri nel CloudFormation modello AWS

Operazione	Description	Competenze richieste
Assegna un nome al bucket S3.	Inserisci il nome del bucket S3 che hai creato nella prima epic.	Architetto del cloud
Fornisci la chiave Amazon S3.	<directory><file-name>Fornisci la posizione del file.zip del codice Lambda nel tuo bucket S3, senza barre iniziali (ad esempio, /.zip).	Architetto del cloud
Fornisci un indirizzo email.	Fornisci un indirizzo e-mail attivo per ricevere le notifiche di Amazon SNS.	Architetto del cloud
Definisci il livello di registrazione.	Definisci il livello e la frequenza di registrazione per la tua funzione Lambda. «Info» indica messaggi informativi dettagliati sullo stato di avanzamento dell'applicazione. «Errore» indica gli eventi di errore che potrebbero comunque consentire all'applicazione di continuare a funzionare. «Avviso» indica situazioni potenzialmente dannose.	Architetto del cloud

Confermare la sottoscrizione

Operazione	Description	Competenze richieste
Confermare la sottoscrizione.	Quando il modello viene distribuito correttamente, invia un messaggio e-mail di sottoscrizione all'indirizzo e-mail fornito. È necessario confermare questa sottoscrizione e-mail per ricevere le notifiche di violazione.	Architetto del cloud

Risorse correlate

- AWS Lambda
- Registrazione di Amazon EMR
- Implementazione di modelli AWS CloudFormation

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Genera dati di test utilizzando un job AWS Glue e Python

Moinul Al-Mamun, Amazon Web Services

Riepilogo

Questo modello mostra come generare in modo rapido e semplice milioni di file di esempio contemporaneamente creando un job AWS Glue scritto in Python. I file di esempio sono archiviati in un bucket Amazon Simple Storage Service (Amazon S3). La capacità di generare rapidamente un gran numero di file di esempio è importante per testare o valutare i servizi nel cloud AWS. Ad esempio, puoi testare le prestazioni dei DataBrew job AWS Glue Studio o AWS Glue eseguendo l'analisi dei dati su milioni di file di piccole dimensioni in un prefisso Amazon S3.

Sebbene sia possibile utilizzare altri servizi AWS per generare set di dati di esempio, consigliamo di utilizzare AWS Glue. Non è necessario gestire alcuna infrastruttura perché AWS Glue è un servizio di elaborazione dati senza server. Basta importare il codice ed eseguirlo in un cluster AWS Glue. Inoltre, AWS Glue fornisce, configura e ridimensiona le risorse necessarie per eseguire i tuoi lavori. Paghi solo per le risorse che le tue attività utilizzano durante l'esecuzione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- AWS Command Line Interface (AWS CLI), installata e configurata per funzionare con l'account AWS

Versioni del prodotto

- Python 3.9
- AWS CLI versione 2

Limitazioni

Il numero massimo di lavori AWS Glue per trigger è 50. Per ulteriori informazioni, consulta gli endpoint e le quote di AWS Glue.

Architecture

Il diagramma seguente mostra un'architettura di esempio incentrata su un job AWS Glue che scrive il suo output (ovvero file di esempio) in un bucket S3.

Il diagramma include il seguente flusso di lavoro:

1. Utilizzi l'AWS CLI, la Console di gestione AWS o un'API per avviare il job AWS Glue. L'API o la CLI di AWS consentono di automatizzare la parallelizzazione del job richiamato e di ridurre il tempo di esecuzione per la generazione di file di esempio.
2. Il job AWS Glue genera il contenuto dei file in modo casuale, lo converte in formato CSV e quindi lo archivia come oggetto Amazon S3 con un prefisso comune. Ogni file è inferiore a un kilobyte. Il job AWS Glue accetta due parametri di lavoro definiti dall'utente: `START_RANGE` e`END_RANGE`. È possibile utilizzare questi parametri per impostare i nomi dei file e il numero di file generati in Amazon S3 da ogni processo eseguito. È possibile eseguire più istanze di questo processo in parallelo (ad esempio, 100 istanze).

Tools (Strumenti)

- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.
- AWS Glue è un servizio di estrazione, trasformazione e caricamento (ETL) completamente gestito. Ti aiuta a classificare, pulire, arricchire e spostare i dati in modo affidabile tra archivi di dati e flussi di dati.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.

Best practice

Prendi in considerazione le seguenti best practice di AWS Glue durante l'implementazione di questo modello:

- Usa il tipo di worker AWS Glue giusto per ridurre i costi. Ti consigliamo di comprendere le diverse proprietà dei tipi di worker e quindi di scegliere il tipo di worker giusto per il tuo carico di lavoro in base ai requisiti di CPU e memoria. Per questo modello, si consiglia di utilizzare un job shell Python come tipo di lavoro per ridurre al minimo la DPU e ridurre i costi. Per ulteriori informazioni, consulta Aggiungere lavori in AWS Glue nella AWS Glue Developer Guide.
- Usa il giusto limite di concorrenza per scalare il tuo lavoro. Ti consigliamo di basare la massima contemporaneità del tuo lavoro AWS Glue sul tempo richiesto e sul numero di file richiesto.
- Inizia a generare un numero limitato di file all'inizio. Per ridurre i costi e risparmiare tempo durante la creazione dei job AWS Glue, inizia con un numero limitato di file (ad esempio 1.000). Questo può semplificare la risoluzione dei problemi. Se la generazione di un numero ridotto di file ha esito positivo, è possibile passare a un numero maggiore di file.
- Esegui prima localmente. Per ridurre i costi e risparmiare tempo nella creazione dei job AWS Glue, avvia lo sviluppo localmente e testa il codice. Per istruzioni sulla configurazione di un contenitore Docker che può aiutarti a scrivere lavori di estrazione, trasformazione e caricamento (ETL) di AWS Glue sia in una shell che in un ambiente di sviluppo integrato (IDE), consulta il post Developing AWS Glue ETL Developing AWS Glue localmente usando un container sul blog di AWS Big Data.

Per ulteriori best practice di AWS Glue, consulta Best practice nella documentazione di AWS Glue.

Epiche

Crea un bucket S3 di destinazione e un ruolo IAM

Operazione	Description	Competenze richieste
Crea un bucket S3 per archiviare i file.	Crea un bucket S3 e un prefisso al suo interno. **Note** Questo modello utilizza la `s3://{your-s3-bucket-name}/small-files/` posizione a scopo dimostrativo.	Sviluppatore di app

Operazione	Description	Competenze richieste
Crea e configura un ruolo IAM.	È necessario creare un ruolo IAM che il job AWS Glue possa utilizzare per scrivere nel bucket S3. 1. Crea un ruolo IAM (ad esempio, chiamato `"AWSGlueServiceRole-smallfiles"`). 2. Scegli AWS Glue come entità affidabile della policy. 3. Allega una policy gestita da AWS chiamata `"AWSGlueServiceRole"` al ruolo. 4. Crea una policy in linea o una policy gestita dal cliente richiamata in `"s3-small-file-access"` base alla seguente configurazione. `"{bucket}"` Sostituiscilo con il nome del tuo bucket. `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [ "s3:GetObject",`	Sviluppatore di app

Operazione	Description	Competenze richieste

	<pre>"s3:PutObject"
	            ],
	            "Resource
	": [
	
	"arn:aws:s3:::{buc
	ket}/small-files/i
	nput/*"
	            ]
	        }
	    ]
	}</pre>
	5. Allega la `"s3-small-file-access"` policy al tuo ruolo.	

Crea e configura un job AWS Glue per gestire esecuzioni simultanee

Operazione	Description	Competenze richieste
Crea un job AWS Glue.	Devi creare un job AWS Glue che generi i tuoi contenuti e li memorizzi in un bucket S3. Crea un lavoro AWS Glue, quindi configura il tuo lavoro completando i seguenti passaggi: 1. Accedi alla Console di gestione AWS e apri la console AWS Glue. 2. Nel pannello di navigazione, in Data Integration and ETL, scegli Jobs.	Sviluppatore di app

Operazione	Description	Competenze richieste
	3. Nella sezione Crea lavoro, scegli l'editor di script Python Shell. 4. Nella sezione Opzioni, seleziona Crea un nuovo script con codice boilerpla te, quindi scegli Crea. 5. Scegli Dettagli del lavoro. 6. Per Nome, inserisci create_small_files. 7. Per IAM Role, seleziona il ruolo IAM che hai creato in precedenza. 8. Nella sezione Questo processo viene eseguito, scegli Un nuovo script da creare da te. 9. Espandi Proprietà avanzate. 10. Per Concorrenza massima, inserire 100 a scopo dimostrativo. Nota: la concorrenza massima definisce il numero di istanze del processo che è possibile eseguire in parallelo. 11. Scegli Save (Salva).	

Operazione	Description	Competenze richieste

Aggiorna il codice del lavoro.	1. Apri la console AWS Glue.
	2. Nel riquadro di navigazione scegliere Jobs (Processi).
	3. Nella sezione I tuoi lavori, scegli il lavoro che hai creato in precedenza.
	4. Scegli la scheda Script, quindi aggiorna lo script in base al codice seguente. Aggiorna le `text_str` variabili `BUCKET_NAME` `PREFIX`, e con i tuoi valori.
	`from awsglue.utils import getResolvedOptions`
	`import sys`
	`import boto3`
	`from random import randrange`
	
	`# Two arguments`
	`args = getResolvedOptions(sys.argv, ['START_RANGE', 'END_RANGE'])`
	
	`START_RANGE = int(args['START_RANGE'])`
	`END_RANGE = int(args['END_RANGE'])`
	
	`BUCKET_NAME = '{BUCKET_NAME}'`
	`PREFIX = 'small-files/input/'`
		Sviluppatore di app

Operazione	Description	Competenze richieste

	s3 = boto3.res
	ource('s3')
	for x in range(STA
	RT_RANGE, END_RANGE
	):
	    # generate file
	 name
	    file_name =
	 f"input_{x}.txt"
	    # generate text
	    text_str =
	 str(randrange(1000
	00))+","+str(randr
	ange(100000))+", "
	 + str(randrange(1000
	0000)) + "," +
	 str(randrange(1000
	0))
	    # write in s3
	s3.Object(BUCKE
	T_NAME, PREFIX +
	 file_name).put(Bod
	y=text_str)
	5. Scegli Save (Salva).	

Esegui il job AWS Glue dalla riga di comando o dalla console

Operazione	Description	Competenze richieste
Esegui il job AWS Glue dalla riga di comando.	Per eseguire il tuo job AWS Glue dalla CLI di AWS, esegui il comando seguente utilizzando i tuoi valori: cmd:~$ aws glue start- job-run --job-name create_small_files --arguments '{"--STAR	Sviluppatore di app

Operazione	Description	Competenze richieste
	T_RANGE":"0","--EN D_RANGE":"1000000"}' cmd:~$ aws glue start- job-run --job-name create_small_files --arguments '{"--STAR T_RANGE":"1000000" ,"--END_RANGE":"20 00000"}' **Note** Per istruzioni sull'esec uzione del job AWS Glue dalla Console di gestione AWS, consulta la storia Run the AWS Glue nella Console di gestione AWS in questo modello. **Tip** Ti consigliamo di utilizzare l'AWS CLI per eseguire i job AWS Glue se desideri eseguire più esecuzioni contempor aneamente con parametri diversi, come mostrato	

Operazione	Description	Competenze richieste

	nell'esempio precedente.
	
	Per generare tutti i comandi AWS CLI necessari per generare un numero definito di file utilizzando un determina to fattore di parallelizzazione, esegui il seguente codice bash (utilizzando i tuoi valori):
	
	`# define parameters`
	`NUMBER_OF_FILES=`
	`10000000;`
	`PARALLELIZATION=50;`
	
	`# initialize`
	`_SB=0;`
	
	`# generate commands`
	`for i in $(seq 1`
	` $PARALLELIZATION);`
	`do`
	`  echo aws glue`
	` start-job-run --`
	`job-name create_sm`
	`all_files --argumen`
	`ts "'"'{"--START_RANG`
	`E":"'$(((NUMBER_OF`
	`_FILES/PARALLELIZA`
	`TION) * (i-1) +`
	` _SB))'","--END_RAN`
	`GE":"'$(((NUMBER_O`
	`F_FILES/PARALLELIZ`
	`ATION) * (i)))'"}'"'";`
	`  _SB=1;`
	`done`	

Operazione	Description	Competenze richieste
	Se usi lo script precedente, considera quanto segue: • Lo script semplifica l'invocazione e la generazione di file di piccole dimensioni su larga scala. • Aggiorna `NUMBER_OF_FILES` e `PARALLELIZATION` con i tuoi valori. • Lo script precedente stampa un elenco di comandi da eseguire. Copia questi comandi di output, quindi eseguili nel tuo terminale. • Se desideri eseguire i comandi direttamente dall'interno dello script, rimuovi l'`echo`istruzione nella riga 11. **Note** Per vedere un esempio di output dello script precedente, vedete l'output dello script Shell nella sezione Informazioni aggiuntive di questo modello.	

Operazione	Description	Competenze richieste
Esegui il job AWS Glue nella Console di gestione AWS.	1. Accedi alla Console di gestione AWS e apri la console AWS Glue. 2. Nel pannello di navigazione, in Data Integration and ETL, scegli Jobs. 3. Nella sezione I tuoi lavori, scegli il tuo lavoro. 4. Nella sezione Parametri (opzionale), aggiorna i parametri. 5. Scegli Azione, quindi scegli Esegui processo. 6. Ripeti i passaggi da 3 a 5 tutte le volte che desideri. Ad esempio, per creare 10 milioni di file, ripeti questo processo 10 volte.	Sviluppatore di app
Verifica lo stato del tuo lavoro con AWS Glue.	1. Apri la console AWS Glue. 2. Nel riquadro di navigazione scegliere Jobs (Processi). 3. Nella sezione I tuoi lavori, scegli il lavoro che hai creato in precedenza (ovvero,`create_small_files`). 4. Per informazioni sull'avanzamento e sulla generazione dei tuoi file, consulta le colonne Run ID, Run Status e altre.	Sviluppatore di app

Risorse correlate

Riferimenti

- Registro dei dati aperti su AWS
- Set di dati per l'analisi
- Dati aperti su AWS
- Aggiungere lavori in AWS Glue
- Guida introduttiva a AWS Glue

Guide e pattern

- Le migliori pratiche di AWS Glue
- Load testing applications

Informazioni aggiuntive

Test di benchmarking

Questo modello è stato utilizzato per generare 10 milioni di file utilizzando diversi parametri di parallelizzazione come parte di un test di benchmarking. La tabella seguente mostra i risultati del test:

Parallelizzazione	Numero di file generati dall'esecuzione di un processo	Durata del lavoro	Velocità
10	1.000.000	6 ore, 40 minuti	Molto lento
50	200.000	80 minuti	Moderata
100	100.000	40 minuti	Veloce

Se desideri velocizzare il processo, puoi configurare più esecuzioni simultanee nella configurazione del processo. Puoi facilmente modificare la configurazione del lavoro in base ai tuoi requisiti, ma tieni presente che esiste un limite di quota del servizio AWS Glue. Per ulteriori informazioni, consulta gli endpoint e le quote di AWS Glue.

Output dello script Shell

L'esempio seguente mostra l'output dello script di shell dal job Run the AWS Glue dalla riga di comando in questo modello.

```
user@MUC-1234567890 MINGW64 ~
  $ # define parameters
  NUMBER_OF_FILES=10000000;
  PARALLELIZATION=50;
  # initialize
  _SB=0;

  # generate commands
  for i in $(seq 1 $PARALLELIZATION);
   do
         echo aws glue start-job-run --job-name create_small_files --arguments
 "'"'{"--START_RANGE":"'$(((NUMBER_OF_FILES/PARALLELIZATION)  (i-1) + SB))'","--
ENDRANGE":"'$(((NUMBER_OF_FILES/PARALLELIZATION)  (i)))'"}'"'";
         _SB=1;
   done

  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"0","--END_RANGE":"200000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"200001","--END_RANGE":"400000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"400001","--END_RANGE":"600000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"600001","--END_RANGE":"800000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"800001","--END_RANGE":"1000000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"1000001","--END_RANGE":"1200000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"1200001","--END_RANGE":"1400000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"1400001","--END_RANGE":"1600000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"1600001","--END_RANGE":"1800000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"1800001","--END_RANGE":"2000000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"2000001","--END_RANGE":"2200000"}'
```

```
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"2200001","--END_RANGE":"2400000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"2400001","--END_RANGE":"2600000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"2600001","--END_RANGE":"2800000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"2800001","--END_RANGE":"3000000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"3000001","--END_RANGE":"3200000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"3200001","--END_RANGE":"3400000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"3400001","--END_RANGE":"3600000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"3600001","--END_RANGE":"3800000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"3800001","--END_RANGE":"4000000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"4000001","--END_RANGE":"4200000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"4200001","--END_RANGE":"4400000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"4400001","--END_RANGE":"4600000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"4600001","--END_RANGE":"4800000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"4800001","--END_RANGE":"5000000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"5000001","--END_RANGE":"5200000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"5200001","--END_RANGE":"5400000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"5400001","--END_RANGE":"5600000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"5600001","--END_RANGE":"5800000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"5800001","--END_RANGE":"6000000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"6000001","--END_RANGE":"6200000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"6200001","--END_RANGE":"6400000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"6400001","--END_RANGE":"6600000"}'
```

```
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"6600001","--END_RANGE":"6800000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"6800001","--END_RANGE":"7000000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"7000001","--END_RANGE":"7200000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"7200001","--END_RANGE":"7400000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"7400001","--END_RANGE":"7600000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"7600001","--END_RANGE":"7800000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"7800001","--END_RANGE":"8000000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"8000001","--END_RANGE":"8200000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"8200001","--END_RANGE":"8400000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"8400001","--END_RANGE":"8600000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"8600001","--END_RANGE":"8800000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"8800001","--END_RANGE":"9000000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"9000001","--END_RANGE":"9200000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"9200001","--END_RANGE":"9400000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"9400001","--END_RANGE":"9600000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"9600001","--END_RANGE":"9800000"}'
  aws glue start-job-run --job-name create_small_files --arguments '{"--
START_RANGE":"9800001","--END_RANGE":"10000000"}'

  user@MUC-1234567890 MINGW64 ~
```

DOMANDE FREQUENTI

Quante esecuzioni simultanee o job paralleli devo usare?

Il numero di esecuzioni simultanee e di lavori paralleli dipende dal tempo richiesto e dal numero desiderato di file di test. Ti consigliamo di controllare la dimensione dei file che stai creando.

Innanzitutto, controlla quanto tempo impiega un job AWS Glue per generare il numero di file desiderato. Quindi, usa il numero giusto di esecuzioni simultanee per raggiungere i tuoi obiettivi. Ad esempio, se presumi che 100.000 file impieghino 40 minuti per completare l'esecuzione ma il tempo previsto sia di 30 minuti, devi aumentare l'impostazione di concorrenza per il tuo job AWS Glue.

Che tipo di contenuto posso creare utilizzando questo modello?

È possibile creare qualsiasi tipo di contenuto, ad esempio file di testo con delimitatori diversi (ad esempio, PIPE, JSON o CSV). Questo modello utilizza Boto3 per scrivere su un file e quindi salva il file in un bucket S3.

Di quale livello di autorizzazione IAM ho bisogno nel bucket S3?

È necessario disporre di una policy basata sull'identità che consenta `Write` l'accesso agli oggetti nel bucket S3. Per ulteriori informazioni, consulta Amazon S3: consente l'accesso in lettura e scrittura agli oggetti in un bucket S3 nella documentazione di Amazon S3.

Inserimento conveniente di dati IoT direttamente in Amazon S3 con AWS IoT Greengrass

Sebastian Viviani e Rizwan Syed, Amazon Web Services

Riepilogo

Questo modello mostra come importare in modo conveniente dati Internet of Things (IoT) direttamente in un bucket Amazon Simple Storage Service (Amazon S3) utilizzando un dispositivo AWS IoT Greengrass versione 2. Il dispositivo esegue un componente personalizzato che legge i dati IoT e li salva in una memoria persistente (ovvero un disco o un volume locale). Quindi, il dispositivo comprime i dati IoT in un file Apache Parquet e carica periodicamente i dati su un bucket S3.

La quantità e la velocità dei dati IoT che acquisisci sono limitate solo dalle funzionalità hardware perimetrali e dalla larghezza di banda della rete. Puoi usare Amazon Athena per analizzare in modo conveniente i dati acquisiti. Athena supporta i file compressi di Apache Parquet e la visualizzazione dei dati utilizzando Amazon Managed Grafana.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un edge gateway che funziona su AWS IoT Greengrass versione 2 e raccoglie dati dai sensori (le fonti di dati e il processo di raccolta dei dati esulano dall'ambito di questo modello, ma è possibile utilizzare quasi tutti i tipi di dati dei sensori. Questo modello utilizza un broker MQTT locale con sensori o gateway che pubblicano dati localmente.)
- Componenti, ruoli e dipendenze SDK di AWS IoT Greengrass
- Un componente di gestione dello stream per caricare i dati nel bucket S3
- SDK AWS per Java, SDK AWS JavaScript per o SDK AWS per Python (Boto3) per eseguire APIs

Limitazioni

- I dati in questo modello non vengono caricati in tempo reale nel bucket S3. Esiste un periodo di ritardo ed è possibile configurare il periodo di ritardo. I dati vengono temporaneamente memorizzati nel buffer nel dispositivo periferico e quindi caricati una volta scaduto il periodo.

- L'SDK è disponibile solo in Java, Node.js e Python.

Architecture

Stack tecnologico Target

- Simple Storage Service (Amazon S3)
- AWS IoT Greengrass
- Broker MQTT
- Componente Stream Manager

Architettura Target

Il diagramma seguente mostra un'architettura progettata per importare i dati dei sensori IoT e archiviarli in un bucket S3.

Il diagramma mostra il flusso di lavoro seguente:

1. Gli aggiornamenti di più sensori (ad esempio, temperatura e valvola) vengono pubblicati su un broker MQTT locale.
2. Il compressore di file Parquet sottoscritto a questi sensori aggiorna gli argomenti e riceve questi aggiornamenti.
3. Il compressore di file Parquet memorizza gli aggiornamenti localmente.
4. Trascorso il periodo, i file memorizzati vengono compressi in file Parquet e passati allo stream manager per essere caricati nel bucket S3 specificato.
5. Lo stream manager carica i file Parquet nel bucket S3.

> **Note**
>
> Lo stream manager (`StreamManager`) è un componente gestito. Per esempi di come esportare dati in Amazon S3, consulta Stream manager nella documentazione di AWS IoT Greengrass. Puoi utilizzare un broker MQTT locale come componente o un altro broker come Eclipse Mosquitto.

Tools (Strumenti)

Strumenti AWS

- Amazon Athena è un servizio di query interattivo che ti aiuta ad analizzare i dati direttamente in Amazon S3 utilizzando SQL standard.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS IoT Greengrass è un servizio cloud e di runtime IoT edge open source che ti aiuta a creare, distribuire e gestire applicazioni IoT sui tuoi dispositivi.

Altri strumenti

- Apache Parquet è un formato di file di dati open source orientato alle colonne progettato per l'archiviazione e il recupero.
- MQTT (Message Queuing Telemetry Transport) è un protocollo di messaggistica leggero progettato per dispositivi con limitazioni.

Best practice

Utilizza il formato di partizione corretto per i dati caricati

Non ci sono requisiti specifici per i nomi dei prefissi root nel bucket S3 (ad esempio, `"myAwesomeDataSet/"` or`"dataFromSource"`), ma ti consigliamo di utilizzare una partizione e un prefisso significativi in modo che sia facile comprendere lo scopo del set di dati.

Ti consigliamo inoltre di utilizzare il giusto partizionamento in Amazon S3 in modo che le query vengano eseguite in modo ottimale sul set di dati. Nell'esempio seguente, i dati vengono partizionati in formato HIVE in modo da ottimizzare la quantità di dati analizzati da ciascuna query Athena. Ciò migliora le prestazioni e riduce i costi.

```
s3://<ingestionBucket>/<rootPrefix>/year=YY/month=MM/day=DD/
HHMM_<suffix>.parquet
```

Epiche

Configurare l'ambiente

Operazione	Description	Competenze richieste
Crea un bucket S3.	1. Crea un bucket S3 o usa un bucket esistente. 2. Crea un prefisso significativo per il bucket S3 in cui desideri inserire i dati IoT (ad esempio,). `s3:\\<bucket>\<prefix>` 3. Registra il tuo prefisso per un uso successivo.	Sviluppatore di app
Aggiungi le autorizzazioni IAM al bucket S3.	Per concedere agli utenti l'accesso in scrittura al bucket e al prefisso S3 che hai creato in precedenza, aggiungi la seguente policy IAM al tuo ruolo AWS IoT Greengrass: `{ "Version": "2012-10-17", "Statement": [ { "Sid": "S3DataUpload", "Effect": "Allow", "Action": [ "s3:List*", "s3:Put*" ],`	Sviluppatore di app

Operazione	Description	Competenze richieste
	"Resource": ["arn:aws:s3:::<ingestionBucket>", "arn:aws:s3:::<ingestionBucket>/<prefix>/*"] }] } Per ulteriori informazioni, consulta Creazione di una policy IAM per accedere alle risorse di Amazon S3 nella documentazione di Aurora. Successivamente, aggiorna la policy delle risorse (se necessario) per il bucket S3 per consentire l'accesso in scrittura con i principali AWS corretti.	

Crea e distribuisci il componente AWS IoT Greengrass

Operazione	Description	Competenze richieste
Aggiorna la ricetta del componente.	Aggiorna la configurazione del componente quando crei una distribuzione in base al seguente esempio: {	Sviluppatore di app

Operazione	Description	Competenze richieste

	<pre>"region": "<region>",
	"parquet_period":
	<period>,
	"s3_bucket":
	"<s3Bucket>",
	"s3_key_prefix":
	"<s3prefix>"
	}</pre>
	Sostituisci `<region>` con la tua regione AWS, `<period>` con il tuo intervallo periodico , `<s3Bucket>` con il tuo bucket S3 e `<s3prefix>` con il tuo prefisso.	

Operazione	Description	Competenze richieste
Crea il componente.	Esegui una delle seguenti operazioni: • Create il componente. • Aggiungete il component e alla CI/CD pipeline (se ne esiste uno). Assicurat i di copiare l'artefatto dal repository degli artefatti al bucket di artefatti AWS IoT Greengrass. Quindi, crea o aggiorna il tuo componente AWS IoT Greengrass. • **Note** Aggiungi il broker MQTT come componente o aggiungilo manualmente in un secondo momento. : Questa decisione influisce sullo schema di autenticazione che è possibile utilizzare con il broker. L'aggiunt a manuale di un broker disaccopp ia il broker da AWS IoT Greengrass e abilita qualsiasi schema di autentica	Sviluppatore di app

Operazione	Description	Competenze richieste
	zione supportato dal broker. I component i del broker forniti da AWS hanno schemi di autentica zione predefiniti. Per ulteriori informazi oni, consulta il broker MQTT 3.1.1 (Moquette) e il broker MQTT 5 (EMQX).	
Aggiornate il client MQTT.	Il codice di esempio non utilizza l'autenticazione perché il componente si connette localmente al broker. Se lo scenario è diverso, aggiorna la sezione client MQTT secondo necessità. Inoltre, effettuate le seguenti operazioni: 1. Aggiornate gli argomenti MQTT nell'abbonamento. 2. Aggiorna il parser dei messaggi MQTT secondo necessità, poiché i messaggi provenienti da ciascuna fonte possono differire.	Sviluppatore di app

Aggiungi il componente al dispositivo core AWS IoT Greengrass versione 2

Operazione	Description	Competenze richieste
Aggiorna la distribuzione del dispositivo principale.	Se la distribuzione del dispositivo core AWS IoT Greengrass versione 2 esiste già, rivedi la distribuzione. Se la distribuzione non esiste, crea una nuova distribuzione. Per assegnare al componente il nome corretto, aggiorna la configurazione del gestore dei registri per il nuovo componente (se necessario) in base a quanto segue: `{ "logsUploaderConfiguration": { "systemLogsConfiguration": { ... }, "componentLogsConfigurationMap": { "<com.iot.ingest.parquet>": { "minimumLogLevel": "INFO", "diskSpaceLimit": "20", "diskSpaceLimitUnit": "MB", "deleteLogFileAfterCloudUpload": "false" } ... }`	Sviluppatore di app

Operazione	Description	Competenze richieste
	<pre> }, "periodicUploadInterval": "300" }</pre> Infine, completa la revisione della distribuzione per il tuo dispositivo principale AWS IoT Greengrass.	

Verifica l'inserimento dei dati nel bucket S3

Operazione	Description	Competenze richieste
Controlla i log per il volume AWS IoT Greengrass.	Verifica quanto segue: • Il client MQTT è connesso correttamente al broker MQTT locale. • Il client MQTT è abbonato agli argomenti corretti. • I messaggi di aggiornamento dei sensori stanno arrivando al broker sugli argomenti MQTT. • La compressione del parquet avviene a ogni intervallo periodico.	Sviluppatore di app
Controlla il bucket S3.	Verifica se i dati vengono caricati nel bucket S3. Puoi vedere i file che vengono caricati in ogni momento.	Sviluppatore di app

Operazione	Description	Competenze richieste
	Puoi anche verificare se i dati vengono caricati nel bucket S3 interrogando i dati nella sezione successiva.	

Configurare l'interrogazione da Athena

Operazione	Description	Competenze richieste
Crea un database e una tabella.	1. Crea un database AWS Glue (se necessario). 2. Crea una tabella in AWS Glue manualmente o eseguendo un crawler in AWS Glue.	Sviluppatore di app
Concedi ad Athena l'accesso ai dati.	1. Aggiorna le autorizzazioni per consentire ad Athena di accedere al bucket S3. Per ulteriori informazioni, consulta Accesso granulare a database e tabelle nel catalogo dati di AWS Glue nella documentazione di Athena. 2. Interroga la tabella nel tuo database.	Sviluppatore di app

risoluzione dei problemi

Problema	Soluzione
Il client MQTT non riesce a connettersi	• Convalida le autorizzazioni sul broker MQTT. Se disponi di un broker MQTT di AWS, consulta broker MQTT 3.1.1 (Moquette) e broker MQTT 5 (EMQX). • Convalida le credenziali sul client MQTT. Se disponi di un broker MQTT di AWS, consulta broker MQTT 3.1.1 (Moquette) e broker MQTT 5 (EMQX).
Il client MQTT non riesce a sottoscrivere	Convalida le autorizzazioni sul broker MQTT. Se disponi di un broker MQTT di AWS, consulta broker MQTT 3.1.1 (Moquette) e broker MQTT 5 (EMQX).
I file Parquet non vengono creati	• Verificate che gli argomenti MQTT siano corretti. • Verificate che i messaggi MQTT provenienti dai sensori siano nel formato corretto.
Gli oggetti non vengono caricati nel bucket S3	• Verifica di disporre della connettività Internet e della connettività degli endpoint. • Verifica che la politica delle risorse per il tuo bucket S3 sia corretta. • Verifica le autorizzazioni per il ruolo principal e del dispositivo AWS IoT Greengrass versione 2.

Risorse correlate

- DataFrame(documentazione Pandas)
- Documentazione Apache Parquet (documentazione Parquet)

- Sviluppa componenti AWS IoT Greengrass (Guida per sviluppatori AWS IoT Greengrass, versione 2)
- Distribuisci i componenti di AWS IoT Greengrass sui dispositivi (AWS IoT Greengrass Developer Guide, versione 2)
- Interagisci con dispositivi IoT locali (AWS IoT Greengrass Developer Guide, versione 2)
- Broker MQTT 3.1.1 (Moquette) (Guida per sviluppatori AWS IoT Greengrass, versione 2)
- Broker MQTT 5 (EMQX) (Guida per sviluppatori AWS IoT Greengrass, versione 2)

Informazioni aggiuntive

Analisi dei costi

Il seguente scenario di analisi dei costi dimostra come l'approccio di ingestione dei dati coperto da questo modello può influire sui costi di inserimento dei dati nel cloud AWS. Gli esempi di prezzo in questo scenario si basano sui prezzi al momento della pubblicazione. I prezzi sono soggetti a modifiche. Inoltre, i costi possono variare in base alla regione AWS, alle quote di servizio AWS e ad altri fattori correlati all'ambiente cloud.

Set di segnali di ingresso

Questa analisi utilizza il seguente set di segnali di ingresso come base per confrontare i costi di ingestione dell'IoT con altre alternative disponibili.

Numero di segnali	Frequenza	Dati per segnale
125	25 Hz	8 byte

In questo scenario, il sistema riceve 125 segnali. Ogni segnale è di 8 byte e si verifica ogni 40 millisecondi (25 Hz). Questi segnali possono provenire singolarmente o raggruppati in un payload comune. Hai la possibilità di dividere e comprimere questi segnali in base alle tue esigenze. Puoi anche determinare la latenza. La latenza è il periodo di tempo necessario per ricevere, accumulare e importare i dati.

A scopo di confronto, l'operazione di importazione per questo scenario si basa nella regione `us-east-1` AWS. Il confronto dei costi si applica solo ai servizi AWS. Altri costi, come l'hardware o la connettività, non vengono presi in considerazione nell'analisi.

Confronti dei costi

La tabella seguente mostra il costo mensile in dollari USA (USD) per ogni metodo di ingestione.

Metodo	Costo mensile
AWS SiteWise IoT*	331,77 DOLLARI
AWS IoT SiteWise Edge con pacchetto di elaborazione dati (mantenimento di tutti i dati all'edge)	200 DOLLARI
Regole di AWS IoT Core e Amazon S3 per l'accesso ai dati grezzi	84,54 DOLLARI
Compressione dei file Parquet a livello periferico e caricamento su Amazon S3	0,5 DOLLARI

*I dati devono essere sottoposti a downsampling per rispettare le quote di servizio. Ciò significa che si verifica una perdita di dati con questo metodo.

Metodi alternativi

Questa sezione mostra i costi equivalenti per i seguenti metodi alternativi:

- AWS IoT SiteWise: ogni segnale deve essere caricato in un messaggio individuale. Pertanto, il numero totale di messaggi al mese è di 125 × 25 × 3600 × 24 × 30, ovvero 8,1 miliardi di messaggi al mese. Tuttavia, AWS IoT SiteWise può gestire solo 10 punti dati al secondo per proprietà. Supponendo che i dati vengano sottoposti a downsampling a 10 Hz, il numero di messaggi al mese viene ridotto a 125×10×3600×24×30, ovvero 3,24 miliardi. Se utilizzi il componente Publisher che raggruppa le misurazioni in gruppi di 10 (a 1 USD per milione di messaggi), ottieni un costo mensile di 324 USD al mese. Supponendo che ogni messaggio sia di 8 byte (1 Kb/125), si tratta di 25,92 GB di spazio di archiviazione dati. Ciò aggiunge un costo mensile di 7,77 USD al mese. Il costo totale per il primo mese è di 331,77 USD e aumenta di 7,77 USD ogni mese.
- AWS IoT SiteWise Edge con pacchetto di elaborazione dati, inclusi tutti i modelli e i segnali completamente elaborati sull'edge (ovvero, nessuna ingestione nel cloud): puoi utilizzare il pacchetto di elaborazione dati come alternativa per ridurre i costi e configurare tutti i modelli che vengono calcolati all'edge. Questo può funzionare solo per l'archiviazione e la visualizzazione,

anche se non viene eseguito alcun calcolo reale. In questo caso, è necessario utilizzare un hardware potente per l'edge gateway. C'è un costo fisso di 200 USD al mese.

- Inserimento diretto in AWS IoT Core tramite MQTT e una regola IoT per archiviare i dati grezzi in Amazon S3 — Supponendo che tutti i segnali siano pubblicati in un payload comune, il numero totale di messaggi pubblicati su AWS IoT Core è di 25×3600×24×30, ovvero 64,8 milioni al mese. A 1 USD per milione di messaggi, si tratta di un costo mensile di 64,8 USD al mese. Con 0,15 USD per milione di attivazioni di regole e con una regola per messaggio, si aggiunge un costo mensile di 19,44 USD al mese. Al costo di 0,023 USD per Gb di storage in Amazon S3, ciò aggiunge altri 1,5 USD al mese (in aumento ogni mese per riflettere i nuovi dati). Il costo totale per il primo mese è di 84,54 USD e aumenta di 1,5 USD ogni mese.
- Compressione dei dati all'edge in un file Parquet e caricamento su Amazon S3 (metodo proposto): il rapporto di compressione dipende dal tipo di dati. Con gli stessi dati industriali testati per MQTT, i dati di output totali per un mese intero sono 1,2 Gb. Questo costa 0,03 USD al mese. I rapporti di compressione (utilizzando dati casuali) descritti in altri benchmark sono dell'ordine del 66 percento (più vicini allo scenario peggiore). Il totale dei dati è di 21 Gb e costa 0,5 USD al mese.

Generatore di file Parquet

Il seguente esempio di codice mostra la struttura di un generatore di file Parquet scritto in Python. L'esempio di codice è solo a scopo illustrativo e non funzionerà se incollato nel tuo ambiente.

```
import queue
import paho.mqtt.client as mqtt
import pandas as pd

#queue for decoupling the MQTT thread
messageQueue = queue.Queue()
client = mqtt.Client()
streammanager = StreamManagerClient()

def feederListener(topic, message):
    payload = {
        "topic" : topic,
        "payload" : message,
    }
    messageQueue.put_nowait(payload)

def on_connect(client_instance, userdata, flags, rc):
    client.subscribe("#",qos=0)
```

```
def on_message(client, userdata, message):
    feederListener(topic=str(message.topic),
 message=str(message.payload.decode("utf-8")))


filename = "tempfile.parquet"
streamname = "mystream"
destination_bucket= "amzn-s3-demo-bucket"
keyname="mykey"
period= 60

client.on_connect = on_connect
client.on_message = on_message
streammanager.create_message_stream(
            MessageStreamDefinition(name=streamname,
 strategy_on_full=StrategyOnFull.OverwriteOldestData)
        )


while True:
   try:
       message = messageQueue.get(timeout=myArgs.mqtt_timeout)
   except (queue.Empty):
       logger.warning("MQTT message reception timed out")

   currentTimestamp = getCurrentTime()
   if currentTimestamp >= nextUploadTimestamp:
       df = pd.DataFrame.from_dict(accumulator)
       df.to_parquet(filename)
       s3_export_task_definition = S3ExportTaskDefinition(input_url=filename,
 bucket=destination_bucket, key=key_name)
       streammanager.append_message(streamname,
 Util.validate_and_serialize_to_json_bytes(s3_export_task_definition))
       accumulator = {}
       nextUploadTimestamp += period
   else:
        accumulator.append(message)
```

Avvia un job Spark in un cluster EMR transitorio utilizzando una funzione Lambda

Dhrubajyoti Mukherjee, Amazon Web Services

Riepilogo

Questo modello utilizza l'azione dell' RunJobFlow API Amazon EMR per avviare un cluster transitorio per eseguire un job Spark da una funzione Lambda. Un cluster EMR temporaneo è progettato per terminare non appena il processo è completo o se si verifica un errore. Un cluster transitorio offre risparmi sui costi perché funziona solo durante il periodo di calcolo e offre scalabilità e flessibilità in un ambiente cloud.

Il cluster EMR transitorio viene avviato utilizzando l'API Boto3 e il linguaggio di programmazione Python in una funzione Lambda. La funzione Lambda, scritta in Python, offre la flessibilità aggiuntiva di avviare il cluster quando è necessario.

Per dimostrare il calcolo e l'output di un batch di esempio, questo modello avvierà un job Spark in un cluster EMR da una funzione Lambda ed eseguirà un calcolo in batch con i dati di vendita di esempio di un'azienda fittizia. L'output del job Spark sarà un file con valori separati da virgole (CSV) in Amazon Simple Storage Service (Amazon S3). Il file di dati di input, il file Spark .jar, uno snippet di codice e un CloudFormation modello AWS per un cloud privato virtuale (VPC) e i ruoli AWS Identity and Access Management (IAM) per eseguire il calcolo sono forniti come allegato.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo

Limitazioni

- È possibile avviare un solo job Spark alla volta dal codice.

Versioni del prodotto

- Testato su Amazon EMR 6.0.0

Architecture

Stack tecnologico Target

- Amazon EMR
- AWS Lambda
- Simple Storage Service (Amazon S3)
- Apache Spark

Architettura di destinazione

Automazione e scalabilità

Per automatizzare il calcolo in batch Spark-EMR, puoi utilizzare una delle seguenti opzioni.

- Implementa una EventBridge regola Amazon in grado di avviare la funzione Lambda in una pianificazione cron. Per ulteriori informazioni, consulta Tutorial: Schedule AWS Lambda functions using. EventBridge
- Configura le notifiche degli eventi di Amazon S3 per avviare la funzione Lambda all'arrivo dei file.
- Passa i parametri di input alla funzione AWS Lambda tramite il corpo dell'evento e le variabili di ambiente Lambda.

Tools (Strumenti)

Servizi AWS

- Amazon EMR è una piattaforma di cluster gestita che semplifica l'esecuzione di framework di big data su AWS per elaborare e analizzare grandi quantità di dati.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri strumenti

- Apache Spark è un motore di analisi multilingue per l'elaborazione di dati su larga scala.

Epiche

Crea i ruoli IAM di Amazon EMR e Lambda e il VPC

Operazione	Description	Competenze richieste
Crea i ruoli IAM e il VPC.	Se disponi già dei ruoli IAM di AWS Lambda e Amazon EMR e di un VPC, puoi saltare questo passaggio. Per eseguire il codice, sia il cluster EMR che la funzione Lambda richiedono ruoli IAM. Il cluster EMR richiede anche un VPC con una sottorete pubblica o una sottorete privata con un gateway NAT. Per creare automaticamente tutti i ruoli IAM e un VPC, distribuisci il CloudFormation modello AWS allegato così com'è oppure puoi creare i ruoli e il VPC manualmente come specifica to nella sezione Informazioni aggiuntive.	Architetto del cloud
Nota le chiavi di output CloudFormation del modello AWS.	Dopo che il CloudFormation modello è stato distribui to correttamente, vai alla scheda Outputs nella console CloudFormation AWS. Nota i cinque tasti di output: • `S3Bucket` • `LambdaExecutionRole`	Architetto del cloud

Operazione	Description	Competenze richieste
	• `ServiceRole` • `JobFlowRole` • `Ec2SubnetId` Utilizzerai i valori di queste chiavi quando creerai la funzione Lambda.	

Carica il file Spark .jar

Operazione	Description	Competenze richieste
Carica il file.jar di Spark.	Carica il file Spark .jar nel bucket S3 creato dallo stack AWS. CloudFormation Il nome del bucket è lo stesso della chiave di output. `S3Bucket`	Informazioni generali su AWS

Crea la funzione Lambda per avviare il cluster EMR

Operazione	Description	Competenze richieste
Creazione di una funzione Lambda.	Sulla console Lambda, crea una funzione Lambda Python 3.9+ con un ruolo di esecuzione. La politica del ruolo di esecuzione deve consentire a Lambda di avviare un cluster EMR. (Vedi il CloudFormation modello AWS allegato).	Ingegnere dei dati, ingegnere del cloud
Copia e incolla il codice.	Sostituisci il codice nel `lambda_function.py`	Ingegnere dei dati, ingegnere del cloud

Operazione	Description	Competenze richieste
	file con il codice della sezione Informazioni aggiuntive di questo modello.	
Modifica i parametri nel codice.	Segui i commenti nel codice per modificare i valori dei parametri in modo che corrispondano al tuo account AWS.	Ingegnere dei dati, ingegnere del cloud
Avvia la funzione per avviare il cluster.	Avvia la funzione per avviare la creazione di un cluster EMR transitorio con il file Spark .jar fornito. Eseguirà il job Spark e terminerà automaticamente quando il job sarà completo.	Ingegnere dei dati, ingegnere del cloud
Verificare lo stato del cluster EMR.	Dopo l'avvio, il cluster EMR viene visualizzato nella console Amazon EMR nella scheda Clusters. Eventuali errori durante l'avvio del cluster o l'esecuzione del processo possono essere controllati di conseguenza.	Ingegnere dei dati, ingegnere del cloud

Configura ed esegui la demo di esempio

Operazione	Description	Competenze richieste
Carica il file.jar di Spark.	Scarica il file Spark .jar dalla sezione Allegati e caricalo nel bucket S3.	Ingegnere dei dati, ingegnere del cloud

Operazione	Description	Competenze richieste
Carica il set di dati di input.	Carica il `fake_sales_data.csv` file allegato nel bucket S3.	Ingegnere dei dati, ingegnere del cloud
Incolla il codice Lambda e modifica i parametri.	Copia il codice dalla sezione Strumenti e incolla il codice in una funzione Lambda, sostituendo il file di codice. `lambda_function.py` Modifica i valori dei parametri in modo che corrispondano al tuo account.	Ingegnere dei dati, ingegnere del cloud
Avvia la funzione e verifica l'output.	Dopo che la funzione Lambda ha avviato il cluster con il job Spark fornito, genera un file.csv nel bucket S3.	Ingegnere dei dati, ingegnere del cloud

Risorse correlate

- Costruire Spark
- Apache Spark e Amazon EMR
- Documentazione run_job_flow di Boto3 Docs
- Informazioni e documentazione su Apache Spark

Informazioni aggiuntive

Codice

```
"""
Copy paste the following code in your Lambda function. Make sure to change the
 following key parameters for the API as per your account

-Name (Name of Spark cluster)
-LogUri (S3 bucket to store EMR logs)
```

```
-Ec2SubnetId (The subnet to launch the cluster into)
-JobFlowRole (Service role for EC2)
-ServiceRole (Service role for Amazon EMR)

The following parameters are additional parameters for the Spark job itself. Change the
 bucket name and prefix for the Spark job (located at the bottom).

-s3://your-bucket-name/prefix/lambda-emr/SparkProfitCalc.jar (Spark jar file)
-s3://your-bucket-name/prefix/fake_sales_data.csv (Input data file in S3)
-s3://your-bucket-name/prefix/outputs/report_1/ (Output location in S3)
"""
import boto3

client = boto3.client('emr')


def lambda_handler(event, context):
    response = client.run_job_flow(
        Name='spark_job_cluster',
        LogUri='s3://your-bucket-name/prefix/logs',
        ReleaseLabel='emr-6.0.0',
        Instances={
            'MasterInstanceType': 'm5.xlarge',
            'SlaveInstanceType': 'm5.large',
            'InstanceCount': 1,
            'KeepJobFlowAliveWhenNoSteps': False,
            'TerminationProtected': False,
            'Ec2SubnetId': 'subnet-XXXXXXXXXXXXX'
        },
        Applications=[{'Name': 'Spark'}],
        Configurations=[
            {'Classification': 'spark-hive-site',
             'Properties': {
                 'hive.metastore.client.factory.class':
 'com.amazonaws.glue.catalog.metastore.AWSGlueDataCatalogHiveClientFactory'}
             }
        ],
        VisibleToAllUsers=True,
        JobFlowRole='EMRLambda-EMREC2InstanceProfile-XXXXXXXXX',
        ServiceRole='EMRLambda-EMRRole-XXXXXXXXX',
        Steps=[
            {
                'Name': 'flow-log-analysis',
                'ActionOnFailure': 'TERMINATE_CLUSTER',
```

```
                'HadoopJarStep': {
                    'Jar': 'command-runner.jar',
                    'Args': [
                        'spark-submit',
                        '--deploy-mode', 'cluster',
                        '--executor-memory', '6G',
                        '--num-executors', '1',
                        '--executor-cores', '2',
                        '--class', 'com.aws.emr.ProfitCalc',
                        's3://your-bucket-name/prefix/lambda-emr/SparkProfitCalc.jar',
                        's3://your-bucket-name/prefix/fake_sales_data.csv',
                        's3://your-bucket-name/prefix/outputs/report_1/'
                    ]
                }
            }
        ]
    )
```

Ruoli IAM e creazione di VPC

Per avviare il cluster EMR in una funzione Lambda, sono necessari ruoli VPC e IAM. Puoi configurare i ruoli VPC e IAM utilizzando il CloudFormation modello AWS nella sezione Attachments di questo modello oppure puoi crearli manualmente utilizzando i seguenti link.

I seguenti ruoli IAM sono necessari per eseguire Lambda e Amazon EMR.

Ruolo di esecuzione Lambda

Il ruolo di esecuzione di una funzione Lambda le concede l'autorizzazione ad accedere ai servizi e alle risorse AWS.

Ruolo di servizio per Amazon EMR

Il ruolo Amazon EMR definisce le azioni consentite per Amazon EMR durante il provisioning di risorse e l'esecuzione di attività a livello di servizio che non vengono eseguite nel contesto di un'istanza Amazon Elastic Compute Cloud (Amazon) in esecuzione all'interno di un cluster. EC2 Ad esempio, il ruolo di servizio viene utilizzato per fornire istanze all'avvio di un cluster. EC2

Ruolo di servizio per le istanze EC2

Il ruolo di servizio per EC2 le istanze di cluster (chiamato anche profilo di EC2 istanza per Amazon EMR) è un tipo speciale di ruolo di servizio assegnato a EC2 ogni istanza di un cluster Amazon EMR

all'avvio dell'istanza. I processi applicativi eseguiti su Apache Hadoop assumono questo ruolo per le autorizzazioni di interazione con altri servizi AWS.

Creazione di VPC e sottoreti

Puoi creare un VPC dalla console VPC.

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Esegui la migrazione dei carichi di lavoro Apache Cassandra su Amazon Keyspaces utilizzando AWS Glue

Nikolai Kolesnikov, Karthiga Priya Chandran e Samir Patel, Amazon Web Services

Riepilogo

Questo modello mostra come migrare i carichi di lavoro Apache Cassandra esistenti su Amazon Keyspaces (per Apache Cassandra) utilizzando AWS Glue. CQLReplicator Puoi utilizzarlo CQLReplicator su AWS Glue per ridurre al minimo il ritardo di replica della migrazione dei carichi di lavoro fino a pochi minuti. Scopri anche come usare un bucket Amazon Simple Storage Service (Amazon S3) per archiviare i dati necessari per la migrazione, inclusi file Apache Parquet, file di configurazione e script. Questo modello presuppone che i carichi di lavoro Cassandra siano ospitati su istanze Amazon Elastic Compute Cloud (Amazon EC2) in un cloud privato virtuale (VPC).

Prerequisiti e limitazioni

Prerequisiti

- Cluster Cassandra con una tabella di origine
- Tabella di destinazione in Amazon Keyspaces per replicare il carico di lavoro
- Bucket S3 per archiviare file Parquet intermedi che contengono modifiche incrementali ai dati
- Bucket S3 per archiviare i file e gli script di configurazione del lavoro

Limitazioni

- CQLReplicator su AWS Glue richiede del tempo per fornire unità di elaborazione dati (DPUs) per i carichi di lavoro Cassandra. È probabile che il ritardo di replica tra il cluster Cassandra e lo spazio chiave e la tabella di destinazione in Amazon Keyspaces duri solo pochi minuti.

Architecture

Stack tecnologico di origine

- Apache Cassandra
- DataStax Server

- ScyllaDB

Stack tecnologico Target

- Amazon Keyspaces

Architettura di migrazione

Il diagramma seguente mostra un'architettura di esempio in cui un cluster Cassandra è ospitato su EC2 istanze e distribuito su tre zone di disponibilità. I nodi Cassandra sono ospitati in sottoreti private.

Il diagramma mostra il flusso di lavoro seguente:

1. Un ruolo di servizio personalizzato fornisce l'accesso ad Amazon Keyspaces e al bucket S3.
2. Un job AWS Glue legge la configurazione del lavoro e gli script nel bucket S3.
3. Il job AWS Glue si connette tramite la porta 9042 per leggere i dati dal cluster Cassandra.
4. Il job AWS Glue si connette tramite la porta 9142 per scrivere dati su Amazon Keyspaces.

Tools (Strumenti)

Servizi e strumenti AWS

- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.
- AWS CloudShell è una shell basata su browser che puoi utilizzare per gestire i servizi AWS utilizzando l'AWS Command Line Interface (AWS CLI) e una gamma di strumenti di sviluppo preinstallati.
- AWS Glue è un servizio ETL completamente gestito che ti aiuta a classificare, pulire, arricchire e spostare in modo affidabile i dati tra archivi e flussi di dati.
- Amazon Keyspaces (per Apache Cassandra) è un servizio di database gestito che ti aiuta a migrare, eseguire e scalare i carichi di lavoro Cassandra nel cloud AWS.

Codice

Il codice per questo pattern è disponibile nel repository. GitHub CQLReplicator

Best practice

- Per determinare le risorse AWS Glue necessarie per la migrazione, stima il numero di righe nella tabella Cassandra di origine. Ad esempio, 250.000 righe per 0,25 DPU (2 vCPUs, 4 GB di memoria) con disco da 84 GB.
- Preriscalda le tabelle Amazon Keyspaces prima dell'esecuzione. CQLReplicator Ad esempio, otto CQLReplicator tile (AWS Glue jobs) possono scrivere fino a 22 K WCUs al secondo, quindi il target deve essere preriscaldato a 25-30 K WCUs al secondo.
- Per abilitare la comunicazione tra i componenti di AWS Glue, utilizza una regola di ingresso autoreferenziale per tutte le porte TCP del tuo gruppo di sicurezza.
- Utilizza la strategia di traffico incrementale per distribuire il carico di lavoro di migrazione nel tempo.

Epiche

Implementare CQLReplicator

Operazione	Description	Competenze richieste
Crea uno spazio chiave e una tabella di destinazione.	1. Crea uno spazio di chiavi e una tabella in Amazon Keyspaces. Per ulteriori informazioni sulla capacità di scrittura, consulta Scrivere i calcoli delle unità nella sezione Informazioni aggiuntive di questo modello. È inoltre possibile creare uno spazio di chiavi utilizzando il Cassandra Query Language (CQL). Per ulteriori informazioni, consulta Creare uno spazio di chiavi utilizzando CQL	Proprietario dell'app, amministratore AWS, DBA, sviluppatore di app

Operazione	Description	Competenze richieste
	nella sezione Informazioni aggiuntive di questo modello. **Note** Dopo aver creato la tabella, valuta la possibilità di passare alla modalità di capacità su richiesta per evitare addebiti inutili. 2. Per eseguire l'aggiornamento alla modalità throughput, esegui lo script seguente: `ALTER TABLE target_keyspace.target_table WITH CUSTOM_PROPERTIES = { 'capacity_mode': { 'throughput_mode': 'PAY_PER_REQUEST'} }`	

Operazione	Description	Competenze richieste
Configura il driver Cassandra per connetterti a Cassandra.	Usa il seguente script di configurazione: `Datastax-java-driver { basic.request.consistency = "LOCAL_QUORUM" basic.contact-points = ["127.0.0.1:9042"] advanced.reconnect-on-init = true basic.load-balancing-policy { local-datacenter = "datacenter1" } advanced.auth-provider = { class = PlainTextAuthProvider username = "user-at-sample" password = "S@MPLE=PASSWORD=" } }` **Note** Lo script precedente utilizza lo Spark Cassandra Connector. Per ulteriori informazioni, consulta la configurazione	DBA

Operazione	Description	Competenze richieste
	di riferimento per Cassandra.	

Operazione	Description	Competenze richieste
Configura il driver Cassandra per la connessione ad Amazon Keyspaces.	Usa il seguente script di configurazione: `datastax-java-driver { basic { load-balancing-policy { local-datacenter = us-west-2 } contact-points = ["cassandra.us-west-2.amazonaws.com:9142"] request { page-size = 2500 timeout = 360 seconds consistency = LOCAL_QUORUM } } advanced { control-connection { timeout = 360 seconds } session-leak.threshold = 6 connection { connect-timeout = 360 seconds init-query-timeout = 360 seconds warn-on-init-error = false } auth-provider = { class = software.aws.mcs.auth.SigV4AuthProvider` [...]	DBA

Operazione	Description	Competenze richieste

```
  aws-region = us-
west-2
 }

 ssl-engine-factory {
  class = DefaultSs
lEngineFactory
        }
    }
}
```

> **Note**
>
> Lo script precedente utilizza lo Spark Cassandra Connector. Per ulteriori informazioni, consulta la configurazione di riferimento per Cassandra.

Operazione	Description	Competenze richieste
Crea un ruolo IAM per il job AWS Glue.	Crea un nuovo ruolo di servizio AWS denominato `glue-cassandra-migration` con AWS Glue come entità affidabile. **Note** `glue-cassandra-migration` Dovrebbe fornire l'accesso in lettura e scrittura al bucket S3 e ad Amazon Keyspaces. Il bucket S3 contiene i file.jar, i file di configurazione per Amazon Keyspaces e Cassandra e i file Parquet intermedi. Ad esempio, contiene le, e le politiche gestite. `AWSGlueServiceRole` `AmazonS3FullAccess` `AmazonKeyspacesFullAccess`	AWS DevOps

Operazione	Description	Competenze richieste

Scarica CQLReplicator in AWS CloudShell.	Scarica il progetto nella tua cartella home eseguendo il seguente comando:
	`git clone https://github.com/aws-samples/cql-replicator .git`
	`cd cql-replicator/glue`
	`# Only for AWS CloudShell, the bc package includes bc and dc. Bc is an arbitrary precision numeric processing arithmetic language`
	`sudo yum install bc -y`	

Modificate i file di configurazione di riferimento.	Copia `Cassandra Connector.conf` e `KeyspacesConnector.conf` `../glue/conf` inseriscilo nella cartella del progetto.	AWS DevOps

Operazione	Description	Competenze richieste

Avvia il processo di migrazione.	Il comando seguente inizializza l'ambiente. CQLReplicator L'inizializzazione prevede la copia di artefatti.jar e la creazione di un connettore AWS Glue, un bucket S3, un job AWS Glue, il keyspace e la tabella: `migration ledger`
	
	`cd cql-replicator/glue/bin`
	`./cqlreplicator --state init --sg '"sg-1","sg-2"' \ --subnet "subnet-XXXXXXXXXXXX" \ --az us-west-2a --region us-west-2 \ --glue-iam-role glue-cassandra-migration \ --landing-zone s3://cql-replicator-1234567890-us-west-2`
	
	Questo script include i seguenti parametri:
	
	• `--sg`— I gruppi di sicurezza che consentono l'accesso al cluster Cassandra da AWS Glue e includono la regola di autoreferenziazione in entrata per tutto il traffico
		AWS DevOps

Operazione	Description	Competenze richieste
	• `--subnet`— La sottorete a cui appartiene il cluster Cassandra • `--az`— La zona di disponibilità della sottorete • `--region`— La regione AWS in cui viene distribuito il cluster Cassandra • `--glue-iam-role` — Le autorizzazioni dei ruoli IAM che AWS Glue può assumere quando chiama Amazon Keyspaces e Amazon S3 per tuo conto • `--landing zone`— Un parametro opzionale per riutilizzare un bucket S3 (se non fornisci un valore per il `--landing zone` parametro, il `init` processo proverà a creare un nuovo bucket per archiviare i file di configurazione, gli artefatti .jar e i file intermedi).	

Operazione	Description	Competenze richieste
Convalida la distribuzione.	Dopo aver eseguito il comando precedente, l'account AWS dovrebbe contenere quanto segue: • Il job CQLReplicator AWS Glue e il connettore AWS Glue in AWS Glue • Il bucket S3 che memorizza gli artefatti • Lo spazio chiave di destinazione `migration` e la `ledger` tabella in Amazon Keyspaces	AWS DevOps

Esegui CQLReplicator

Operazione	Description	Competenze richieste
Avvia il processo di migrazion e.	Per operare CQLReplicator su AWS Glue, è necessari o utilizzare il `--state run` comando seguito da una serie di parametri. La configurazione precisa di questi parametri è determina ta principalmente dai tuoi requisiti di migrazione unici. Ad esempio, queste impostazi oni potrebbero variare se scegli di replicare i valori e gli aggiornamenti del time to live (TTL) o se scarichi oggetti	AWS DevOps

Operazione	Description	Competenze richieste
	superiori a 1 MB su Amazon S3. Per replicare il carico di lavoro dal cluster Cassandra ad Amazon Keyspaces, esegui il seguente comando: `./cqlreplicator --state run --tiles 8 \ --landing-zone s3://cql-replicator-1234567890-us-west-2 \ --region us-west-2 \ --src-keyspace source_keyspace \ --src-table source_table \ --trg-keyspace taget_keyspace \ --writetime-column column_name \ --trg-table target_table --inc-traffic` Lo spazio chiave e la tabella di origine si trovano `source_keyspace.source_table` nel cluster Cassandra. Lo spazio chiave e la tabella di destinazione si trovano `target_keyspace.ta`	

Operazione	Description	Competenze richieste
	`rget_table` in Amazon Keyspaces. Il parametro `--inc-traffic` aiuta a evitare che il traffico incrementale sovraccarichi il cluster Cassandra e Amazon Keyspaces con un numero elevato di richieste. Per replicare gli aggiornamenti, `--writetime-column regular_column_name` aggiungili alla riga di comando. La colonna normale verrà utilizzata come fonte del timestamp di scrittura.	

Monitora il processo di migrazione

Operazione	Description	Competenze richieste
Convalida le righe di Cassandra migrate durante la fase di migrazione storica.	Per ottenere il numero di righe replicate durante la fase di riempimento, esegui il seguente comando: `./cqlreplicator --state stats \ --landing-zone s3://cql-replicator-1234567890-us-west-2 \ --src-keyspace source_keyspace --src-table`	AWS DevOps

Operazione	Description	Competenze richieste
	`source_table --region us-west-2`	

Interrompi il processo di migrazione

Operazione	Description	Competenze richieste
Usa il `cqlreplicator` comando o la console AWS Glue.	Per interrompere correttamente il processo di migrazione, esegui il seguente comando: `./cqlreplicator --state request-stop --tiles 8 \ --landing-zone s3://cql-replicator-1234567890-us-west-2 \ --region us-west-2 \ --src-keyspace source_keyspace --src-table source_table` Per interrompere immediatamente il processo di migrazione, utilizza la console AWS Glue.	AWS DevOps

Eliminazione

Operazione	Description	Competenze richieste
Eliminare le risorse distribuite.	Il seguente comando eliminerà il job AWS Glue, il connettore, il bucket S3 e la tabella Keyspaces: `ledger` `./cqlreplicator --state cleanup --landing-zone s3://cql-replicator-1234567890-us-west-2`	AWS DevOps

risoluzione dei problemi

Problema	Soluzione
I job di AWS Glue non sono riusciti e hanno restituito un errore di memoria esaurita (OOM).	1. Cambia il tipo di lavoratore (scalabilità verticale). Ad esempio, cambia `G0.25X` in `G.1X` o `G.1X` in`G.2X`. In alternativa, aumenta il numero di input DPUs per ogni job AWS Glue (con scalabilità orizzontale) CQLReplicator. 2. Avvia il processo di migrazione dal punto in cui è stato interrotto. Per riavviare i CQLReplicator processi non riusciti, esegui nuovamente il `--state run` comando con gli stessi parametri.

Risorse correlate

- CQLReplicator con AWS Glue README.MD
- Documentazione AWS Glue

- Documentazione di Amazon Keyspaces
- Apache Cassandra

Informazioni aggiuntive

Considerazioni sulla migrazione

Puoi utilizzare AWS Glue per migrare il carico di lavoro di Cassandra su Amazon Keyspaces, mantenendo al contempo i database di origine Cassandra completamente funzionanti durante il processo di migrazione. Una volta completata la replica, puoi scegliere di trasferire le tue applicazioni su Amazon Keyspaces con un ritardo di replica minimo (meno di minuti) tra il cluster Cassandra e Amazon Keyspaces. Per mantenere la coerenza dei dati, puoi anche utilizzare una pipeline simile per replicare i dati nel cluster Cassandra da Amazon Keyspaces.

Scrivi calcoli unitari

Ad esempio, considera che intendi scrivere 500.000.000 con la dimensione della riga 1 KB nell'arco di un'ora. Il numero totale di unità di scrittura Amazon Keyspaces (WCUs) richieste si basa su questo calcolo:

```
(number of rows/60 mins 60s) 1 WCU per row = (500,000,000/(60*60s) * 1 WCU)
= 69,444 WCUs required
```

69.444 WCUs al secondo è la tariffa per 1 ora, ma potresti aggiungere un po' di protezione per le spese generali. Ad esempio, `69,444 * 1.10 = 76,388 WCUs` ha spese generali del 10%.

Crea uno spazio chiave utilizzando CQL

Per creare uno spazio chiave utilizzando CQL, esegui i seguenti comandi:

```
CREATE KEYSPACE target_keyspace WITH replication = {'class': 'SingleRegionStrategy'}
CREATE TABLE target_keyspace.target_table ( userid uuid, level text, gameid int,
 description text, nickname text, zip text, email text, updatetime text, PRIMARY KEY
 (userid, level, gameid) ) WITH default_time_to_live = 0 AND CUSTOM_PROPERTIES =
 {'capacity_mode':{ 'throughput_mode':'PROVISIONED', 'write_capacity_units':76388,
 'read_capacity_units':3612 }} AND CLUSTERING ORDER BY (level ASC, gameid ASC)
```

Esegui la migrazione di Oracle Business Intelligence 12c al cloud AWS dai server locali

Showunmi Lanre (Lan-Ray) e Patrick Huang, Amazon Web Services

Riepilogo

Questo modello mostra come migrare Oracle Business Intelligence Enterprise Edition 12c dai server locali al cloud AWS utilizzando AWS. CloudFormation Descrive inoltre come utilizzare altri servizi AWS per implementare componenti Oracle BI 12c che offrono alta disponibilità, sicurezza, flessibilità e capacità di scalabilità dinamica.

Per un elenco di best practice relative alla migrazione di Oracle BI 12c al cloud AWS, consulta la sezione Informazioni aggiuntive di questo modello.

> **Note**
>
> È consigliabile eseguire più migrazioni di test prima di trasferire i dati Oracle BI 12c esistenti sul cloud. Questi test ti aiutano a perfezionare il tuo approccio alla migrazione, a identificare e risolvere potenziali problemi e a stimare i requisiti di inattività con maggiore precisione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Connettività di rete sicura tra i server locali e AWS tramite i servizi AWS Virtual Private Network (AWS VPN) o AWS Direct Connect
- Licenze software per il tuo sistema operativo Oracle, Oracle BI 12c, Oracle Database, Oracle WebLogic Server e Oracle HTTP Server

Limitazioni

Per informazioni sui limiti delle dimensioni di storage, consulta la documentazione di Amazon Relational Database Service (Amazon RDS) per Oracle.

Versioni del prodotto

- Oracle Business Intelligence Enterprise Edition 12c
- Oracle WebLogic Server 12c
- Oracle HTTP Server 12c
- Oracle Database 12c (o versione successiva)
- Oracle Java SE 8

Architecture

Il diagramma seguente mostra un'architettura di esempio per l'esecuzione di componenti Oracle BI 12c nel cloud AWS:

Questo diagramma mostra la seguente architettura:

1. Amazon Route 53 fornisce la configurazione DNS (Domain Name Service).
2. Elastic Load Balancing (ELB) distribuisce il traffico di rete per migliorare la scalabilità e la disponibilità dei componenti di Oracle BI 12c su più zone di disponibilità.
3. I gruppi Amazon Elastic Compute Cloud (Amazon EC2) Auto Scaling ospitano i server HTTP Oracle, il server Weblogic Admin e i server BI gestiti in più zone di disponibilità.
4. Amazon Relational Database Service (Amazon RDS) per database Oracle archivia i metadati del server BI su più zone di disponibilità.
5. Amazon Elastic File System (Amazon EFS) è montato su ogni componente di Oracle BI 12c per lo storage condiviso di file.

Stack tecnologico

- Amazon Elastic Block Store (Amazon EBS)
- Amazon Elastic Compute Cloud (Amazon EC2)
- Amazon Elastic File System (Amazon EFS)
- Amazon RDS per Oracle
- AWS Certificate Manager (ACM)
- Elastic Load Balancing (ELB)
- Oracle BI 12c
- Oracle WebLogic Server 12c

- Server HTTP Oracle (OHS)

Tools (Strumenti)

- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account e regioni AWS.
- AWS Certificate Manager (ACM) ti aiuta a creare, archiviare e rinnovare certificati e chiavi SSL/TLS X.509 pubblici e privati che proteggono i tuoi siti Web e le tue applicazioni AWS.
- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e scalarli rapidamente verso l'alto o verso il basso.
- Amazon EC2 Auto Scaling ti aiuta a mantenere la disponibilità delle applicazioni e ti consente di aggiungere o rimuovere automaticamente EC2 istanze Amazon in base alle condizioni da te definite.
- Amazon Elastic File System (Amazon EFS) ti aiuta a creare e configurare file system condivisi nel cloud AWS.
- Elastic Load Balancing distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni. Ad esempio, puoi distribuire il traffico tra istanze Amazon Elastic Compute Cloud (Amazon EC2), contenitori e indirizzi IP in una o più zone di disponibilità.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito. Questa rete virtuale è simile a una rete tradizionale che gestiresti nel tuo data center, con i vantaggi dell'utilizzo dell'infrastruttura scalabile di AWS.
- Oracle Data Pump ti aiuta a spostare dati e metadati da un database all'altro a velocità elevate.
- Oracle Fusion Middleware è una suite di strumenti di sviluppo di applicazioni e soluzioni di integrazione per la gestione delle identità, la collaborazione e la reportistica di business intelligence.
- Oracle ti GoldenGate aiuta a progettare, eseguire, orchestrare e monitorare la replica dei dati e le soluzioni di elaborazione dei dati in streaming nell'infrastruttura Oracle Cloud.

- Oracle WebLogic Scripting Tool (WLST) fornisce un'interfaccia a riga di comando che consente di scalare orizzontalmente i cluster. WebLogic

Epiche

Valuta l'ambiente di origine

Operazione	Description	Competenze richieste
Raccogli informazioni sull'inventario del software.	Identifica le versioni e i livelli di patch per ciascuno dei componenti software del tuo stack tecnologico di origine, inclusi i seguenti: • Sistema operativo Oracle • database Oracle • Oracle BI 12c • Server Oracle WebLogic • Server HTTP Oracle • Java	Architetto della migrazione, architetto delle soluzioni, proprietario dell'applicazione, amministratore di Oracle BI
Raccogli informazioni sull'inventario di calcolo e storage.	Nel tuo ambiente di origine, esamina le metriche di utilizzo attuali e storiche per quanto segue: • Utilizzo CPU • Utilizzo della memoria • Utilizzo dello storage	Architetto della migrazione, architetto delle soluzioni, proprietario dell'applicazione, amministratore di Oracle BI, amministratore di sistema

Operazione	Description	Competenze richieste
	Important Assicurati di tenere conto dei picchi storici di utilizzo.	
Raccogli informazioni sull'architettura dell'ambiente di origine e sui relativi requisiti.	Acquisite una conoscenza completa dell'architettura dell'ambiente di origine e dei relativi requisiti, inclusa la conoscenza di quanto segue: • Configurazione WebLogic del dominio Oracle Server • Clustering • Sistema di bilanciamento del carico • Connettività • Disponibilità • Requisiti per il disaster recovery	Architetto della migrazione, architetto delle soluzioni, proprietario dell'applicazione, amministratore di Oracle BI
Identifica le fonti di dati Java Database Connectivity (JDBC).	Raccogli informazioni sulle fonti di dati e sui driver JDBC del tuo ambiente di origine per ogni motore di database che utilizza.	Architetto della migrazione, proprietario dell'applicazione, amministratore di Oracle BI, ingegnere o amministratore del database

Operazione	Description	Competenze richieste
Raccogli informazioni sulle impostazioni specifiche dell'ambiente.	Raccogli informazioni su impostazioni e configurazioni specifiche dell'ambiente di origine, tra cui: • Script di avvio e spegnimento personalizzati • Java e altre variabili di ambiente • Certificati	Architetto della migrazione, architetto delle soluzioni, proprietario dell'applicazione, amministratore di Oracle BI
Identifica eventuali dipendenze da altre applicazioni.	Raccogli informazioni sulle integrazioni nel tuo ambiente di origine che creano dipendenze con altre applicazioni. **Important** Assicurati di identificare eventuali integrazioni LDAP (Lightweight Directory Access Protocol) e altri requisiti di rete.	Architetto della migrazione, architetto delle soluzioni, proprietario dell'applicazione, amministratore di Oracle BI

Progetta il tuo ambiente di destinazione

Operazione	Description	Competenze richieste
Crea un documento di progettazione di alto livello.	Crea un documento di progettazione architettonica di destinazione. Assicuratevi di utilizzare le informazioni	Architetto delle soluzioni, architetto delle applicazioni, ingegnere del database, architetto della migrazione

Operazione	Description	Competenze richieste
	raccolte durante la valutazio ne dell'ambiente di origine per elaborare il documento di progettazione.	
Ottenere l'approvazione per il documento di progettazione.	Rivedi il documento di progettazione con le parti interessate e ottieni le approvazioni richieste.	Proprietario dell'applicazione o del servizio, architetto delle soluzioni, architetto dell'appl icazione

Implementa l'infrastruttura

Operazione	Description	Competenze richieste
Prepara il codice dell'infr astruttura in CloudFormation.	Crea CloudFormation modelli per effettuare il provisioning della tua infrastruttura Oracle BI 12c nel cloud AWS. Per ulteriori informazioni, consulta Working with AWS CloudFormation templates nella AWS CloudFormation User Guide. **Note** È consigliabile creare CloudFormation modelli modulari per ogni livello di Oracle BI 12c, anziché un modello di grandi dimensioni per tutte le risorse. Per ulteriori	Architetto dell'infrastruttura cloud, architetto delle soluzioni , architetto delle applicazioni

Operazione	Description	Competenze richieste
	informazioni sulle CloudFormation best practice, consulta 8 best practice per automatizzare le distribuzioni con AWS CloudFormation sul blog AWS.	
Scarica il software richiesto.	DDownload il seguente software insieme alle versioni e alle patch richieste dal sito Web di Oracle: • Java JDK8 • Oracle WebLogic Server 12c • Oracle BI 12c	Architetto della migrazione, ingegnere del database, architetto delle applicazioni
Preparare gli script di installazione.	Crea script di installazione software che eseguano un'installazione invisibile all'utente. Questi script semplificano l'automazione della distribuzione. Per ulteriori informazioni, vedere OBIEE 12c: Come eseguire un'installazione silenziosa? sul sito Oracle Support. È necessario un account Oracle Support per visualizzare la documentazione.	Architetto della migrazione, ingegnere del database, architetto dell'applicazione

Operazione	Description	Competenze richieste
Crea un'AMI Linux supportata da Amazon EBS per i tuoi livelli web e applicativi.	1. Distribuisci e configura EC2 istanze Amazon per i tuoi livelli web e applicativi. Assicurati che le istanze soddisfino i prerequisiti per l'esecuzione di quanto segue: • Configurazione dell'ambiente del sistema operativo Oracle • Configurazione dell'account utente del sistema operativo Oracle • Installazione del software Java 2. Crea Amazon Machine Images (AMIs) delle istanze e salva copie per utilizzi futuri. Per istruzioni, consulta Creare un'AMI Linux supportata da Amazon EBS nella Amazon EC2 User Guide for Linux Instances.	Architetto della migrazione, ingegnere di database, architetto delle applicazioni
Avvia la tua infrastruttura AWS utilizzando CloudFormation.	Distribuisci i livelli web e applicativi di Oracle BI 12c in moduli utilizzando i CloudFormation modelli che hai creato. Per istruzioni, consulta Getting started with AWS CloudFormation nella AWS CloudFormation User Guide.	Architetto dell'infrastruttura cloud, architetto delle soluzioni, architetto delle applicazioni

Esegui la migrazione di Oracle BI 12c ad AWS utilizzando una nuova installazione

Operazione	Description	Competenze richieste
Prepara il software richiesto.	Posiziona il software richiesto in una posizione accessibile alle EC2 istanze Amazon. Ad esempio, puoi eseguire lo stage del software in Amazon S3 o in un'altra EC2 istanza Amazon accessibile ai tuoi server Web e applicativi.	Architetto della migrazione, architetto di Oracle BI, architetto dell'infrastruttura cloud, architetto delle soluzioni, architetto delle applicazioni
Prepara il database del repository per l'installazione di Oracle BI 12c.	Crea schemi Oracle BI 12c eseguendo l'Oracle Repository Creation Utility (RCU) su una nuova istanza di database Amazon RDS for Oracle.	Architetto dell'infrastruttura cloud, architetto delle soluzioni, architetto delle applicazioni, architetto della migrazione, architetto di Oracle BI
Installa Oracle Fusion Middleware 12c e Oracle BI 12c.	1. A partire da un' EC2 istanza Amazon, installa l'infrastruttura Oracle Fusion Middleware 12c e OBIEE 12c. Per ulteriori informazioni, consulta le seguenti sezioni della Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence: • Avvio del programma di installazione dell'infrastruttura su BIHOST1 • Installazione di Oracle Business Intelligence in preparazione di una distribuzione aziendale	Architetto della migrazione, Oracle BI Architect

Operazione	Description	Competenze richieste
	Note Usa Amazon EFS per ospitare le directory che verranno condivise tra i nodi del cluster Oracle BI 12c. 2. Applica tutte le patch necessarie all'installazione. 3. Crea AMIs delle istanze e salva le copie per usi futuri.	
Configura il tuo dominio Oracle WebLogic Server per Oracle BI 12c.	Configura il tuo dominio Oracle BI 12c come distribuzione non in cluster. Per ulteriori informazioni, vedere Configurazione del dominio BI nella Oracle Fusion Middleware Enterprise Deployment Guide per Oracle Business Intelligence.	Architetto della migrazione, Oracle BI Architect

Operazione	Description	Competenze richieste
Esegui la scala orizzontale con Oracle BI 12c.	Ridimensiona orizzontalmente il singolo nodo fino al numero di nodi desiderato. Per ulteriori informazioni, vedere Scaling out Oracle Business Intelligence nella Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence.	Architetto della migrazione, Oracle BI Architect
Installare Oracle HTTP Server 12c.	1. Installa Oracle HTTP Server 12c sulle EC2 istanze Amazon di livello Web Oracle. Per istruzioni, vedere Installare Oracle HTTP Server 12c in Installare e configurare Oracle HTTP Server per Oracle Access Management 12c. 2. Applica tutte le patch necessarie all'installazione. 3. Crea AMIs delle istanze e salva le copie per usi futuri.	Architetto della migrazione, Oracle BI Architect
Configura i sistemi di bilanciamento del carico per la terminazione SSL.	1. Crea o importa certificati SSL in ACM. 2. Associa i certificati SSL a ELB.	Architetto dell'infrastruttura cloud, architetto della migrazione

Operazione	Description	Competenze richieste
Migra gli artefatti dei metadati di business intelligence su AWS.	1. Esporta i file Oracle Business Intelligence Application Archive (BAR) dall'installazione locale di Oracle BI 12c. Per esportare i file BAR, utilizzare lo WebLogic Scripting Tool (WLST) per eseguire il comando. exportServiceInstance 2. Importa i file BAR locali nell'installazione di AWS Oracle BI 12c. Per importare i file BAR, esegui il comando importSer viceInstanceWLST.	Architetto della migrazione, Oracle BI Architect

Operazione	Description	Competenze richieste
Esegui attività successive alla migrazione.	Dopo aver importato i file BAR, effettuate le seguenti operazioni: • Configura eventuali sorgenti dati JDBC aggiuntive. • Installa driver per altre fonti di dati come PostgreSQL o Amazon Redshift. • Configura Oracle LDAP, SSL, Single Sign-On (SSO) e Security Store. WebLogic • Configura le policy di AWS Identity and Access Management (IAM). • Attiva il monitoraggio dell'util izzo. • Configura integrazioni con altri sistemi. • Esegui la migrazione di qualsiasi script personali zzato.	Architetto della migrazione, Oracle BI Architect

Prova il nuovo ambiente

Operazione	Description	Competenze richieste
Prova il nuovo ambiente Oracle BI 12c.	Esegui end-to-end test sul nuovo ambiente Oracle BI 12c. Usa l'automazione il più possibile.	Architetto della migrazion e, architetto delle soluzioni, proprietario dell'applicazione, amministratore di Oracle BI

Operazione	Description	Competenze richieste
	Alcuni esempi di attività di test includono quanto segue: • Convalida di dashboard, report e URLs • Test di accettazione degli utenti (UAT) • Test di accettazione operativa (OAT) **Note** Effettuare test e convalide aggiuntivi, se necessario.	

Passa al nuovo ambiente

Operazione	Description	Competenze richieste
Disconnetti il traffico all'ambiente Oracle BI 12c locale.	Alla finestra di apertura stabilita, interrompi tutto il traffico verso l'ambiente Oracle BI 12c locale.	Architetto della migrazione, architetto delle soluzioni, proprietario dell'applicazione, amministratore di Oracle BI
Risincronizza il nuovo database del repository Oracle BI 12c con il database di origine.	Risincronizza il database del repository Amazon RDS Oracle Oracle BI 12c con il database locale. Per sincronizzare i database, puoi utilizzare un aggiornamento di Oracle Data Pump o	Amministratore Oracle BI, ingegnere/amministratore del database

Operazione	Description	Competenze richieste
	un CDC (change data capture) di AWS DMS.	
Passa il tuo Oracle BI 12c URLs in modo che punti al nuovo ambiente AWS.	Aggiorna Oracle BI 12c URLs sui tuoi server DNS interni in modo che puntino alla nuova installazione AWS.	Architetto della migrazione, architetto delle soluzioni, proprietario dell'applicazione, amministratore di Oracle BI
Monitora il nuovo ambiente.	Monitora il nuovo ambiente Oracle BI 12c utilizzando uno dei seguenti strumenti: • Amazon CloudWatch • Approfondimenti sulle prestazioni di Amazon RDS • Oracle Enterprise Manager	Amministratore Oracle BI, ingegnere/amministratore del database, amministratore delle applicazioni
Ottieni l'approvazione del progetto.	Rivedi i risultati dei test con le parti interessate e ottieni le approvazioni necessarie per concludere la migrazione.	Proprietario dell'applicazione, proprietario del servizio, architetto dell'infrastruttura cloud, architetto della migrazione, architetto Oracle BI

Risorse correlate

- Utilizzo della Oracle Repository Creation Utility su RDS per Oracle (Amazon RDS User Guide)
- Oracle su Amazon RDS (Guida per l'utente di Amazon RDS)
- Oracle WebLogic Server 12c su AWS (white paper AWS)
- Implementazione di Oracle Business Intelligence per l'alta disponibilità (Oracle Help Center)
- File Oracle Business Intelligence Application Archive (BAR) (Centro assistenza Oracle)
- Come migrare OBI 12c tra ambienti (Oracle Support)

Informazioni aggiuntive

Di seguito è riportato un elenco di best practice relative alla migrazione di Oracle BI 12c al cloud AWS.

Database di repository

È consigliabile ospitare gli schemi di database Oracle BI 12c su un'istanza Amazon RDS for Oracle. Questo tipo di istanza offre una capacità ridimensionabile e conveniente, automatizzando al contempo le attività di amministrazione, come il provisioning dell'hardware, la configurazione del database, l'applicazione di patch e i backup.

Per ulteriori informazioni, consulta Using the Oracle Repository Creation Utility on RDS for Oracle nella Amazon RDS User Guide.

Livelli Web e applicativi

Le EC2 istanze Amazon ottimizzate per la memoria sono spesso adatte per i server Oracle BI 12c. Qualunque sia il tipo di istanza scelto, assicurati che le istanze di cui effettui il provisioning soddisfino i requisiti di utilizzo della memoria del sistema. Inoltre, assicurati di configurare una dimensione dell'heap WebLogic Java Virtual Machine (JVM) sufficiente in base alla memoria disponibile dell' EC2 istanza Amazon.

Archiviazione locale

L'I/O svolge un ruolo importante nelle prestazioni complessive dell'applicazione Oracle BI 12c. Amazon Elastic Block Store (Amazon EBS) offre diverse classi di storage ottimizzate per diversi modelli di carico di lavoro. Assicurati di scegliere un tipo di volume Amazon EBS adatto al tuo caso d'uso.

Per ulteriori informazioni sui tipi di volume EBS, consulta le caratteristiche di Amazon EBS nella documentazione di Amazon EBS.

Archiviazione condivisa

Un dominio Oracle BI 12c in cluster richiede uno storage condiviso per le seguenti risorse:

- File di configurazione
- Directory di dati singleton (SDD) di Oracle BI 12c
- Cache globale Oracle

- Script di Oracle BI Scheduler
- File binari di Oracle Server WebLogic

Puoi soddisfare questo requisito di storage condiviso utilizzando Amazon EFS, che fornisce un file system NFS (Network File System) elastico scalabile e completamente gestito.

Ottimizzazione delle prestazioni di storage condiviso

Amazon EFS offre due modalità di throughput: Provisioned e Bursting. Il servizio offre anche due modalità di prestazioni: General Purpose e Max I/O.

Per ottimizzare le prestazioni, inizia testando i carichi di lavoro nella modalità prestazioni General Purpose e nella modalità Provisioned throughput. L'esecuzione di questi test ti aiuterà a determinare se tali modalità di base sono sufficienti a soddisfare i livelli di servizio desiderati.

Per ulteriori informazioni, consulta le prestazioni di Amazon EFS nella Guida per l'utente di Amazon EFS.

Disponibilità e disaster recovery

È consigliabile distribuire i componenti di Oracle BI 12c su più zone di disponibilità per proteggere tali risorse in caso di guasto di una zona di disponibilità. Di seguito è riportato un elenco di best practice di disponibilità e disaster recovery per specifiche risorse Oracle BI 12c ospitate nel cloud AWS:

- Database di repository Oracle BI 12c: distribuisci un'istanza di database Amazon RDS Multi-AZ nel tuo database di repository Oracle BI 12c. In una distribuzione Multi-AZ, Amazon RDS effettua automaticamente il provisioning e mantiene una replica sincrona in standby in una zona di disponibilità diversa. L'esecuzione di un'istanza di database di repository Oracle BI 12c nelle zone di disponibilità può migliorare la disponibilità durante la manutenzione pianificata del sistema e aiutare a proteggere i database dai guasti delle istanze e delle zone di disponibilità.
- Server gestiti Oracle BI 12c: per ottenere la tolleranza agli errori, è consigliabile distribuire i componenti di sistema Oracle BI 12c sui server gestiti in un gruppo Amazon EC2 Auto Scaling configurato per estendersi su più zone di disponibilità. Auto Scaling sostituisce le istanze difettose sulla base dei controlli sanitari di Amazon. EC2 In caso di guasto della zona di disponibilità, i server HTTP Oracle continuano a indirizzare il traffico verso i server gestiti nella zona di disponibilità funzionante. Quindi, Auto Scaling avvia le istanze per tenere il passo con i requisiti di numero di host. Si consiglia di attivare la replica dello stato della sessione HTTP per garantire un failover regolare delle sessioni esistenti sui server gestiti funzionanti.

- Administration Server Oracle BI 12c: per assicurarti che il tuo Administration Server abbia un'elevata disponibilità, ospitalo in un gruppo Amazon EC2 Auto Scaling configurato per coprire più zone di disponibilità. Quindi, imposta la dimensione minima e massima del gruppo su 1. Se si verifica un errore nella zona di disponibilità, Amazon EC2 Auto Scaling avvia un Administration Server sostitutivo in una zona di disponibilità alternativa. Per ripristinare eventuali host sottostanti guasti all'interno della stessa zona di disponibilità, puoi attivare Amazon EC2 Auto Recovery.
- Server Oracle Web Tier: è consigliabile associare il server HTTP Oracle al dominio Oracle WebLogic Server. Per un'elevata disponibilità, distribuisci il tuo Oracle HTTP Server in un gruppo Amazon EC2 Auto Scaling configurato per coprire più zone di disponibilità. Quindi, posiziona il server dietro un sistema di bilanciamento del carico elastico ELB. Per fornire una protezione aggiuntiva contro i guasti dell'host, puoi attivare Amazon EC2 Auto Recovery.

Scalabilità

L'elasticità del cloud AWS ti aiuta a scalare le applicazioni orizzontalmente o verticalmente in risposta ai requisiti del carico di lavoro.

Scalabilità verticale

Per scalare verticalmente la tua applicazione, puoi modificare la dimensione e il tipo delle EC2 istanze Amazon che eseguono i componenti di Oracle BI 12c. Non è necessario sovradimensionare le istanze all'inizio della distribuzione e incorrere in costi inutili.

Scalabilità orizzontale

Amazon EC2 Auto Scaling ti aiuta a scalare orizzontalmente la tua applicazione aggiungendo o rimuovendo automaticamente server gestiti in base ai requisiti del carico di lavoro.

> **Note**
>
> La scalabilità orizzontale con Amazon EC2 Auto Scaling richiede competenze di scripting e test approfonditi per essere implementata.

Backup e ripristino

Di seguito è riportato un elenco di best practice di backup e ripristino per specifiche risorse Oracle BI 12c ospitate nel cloud AWS:

- Archivi di metadati di Oracle Business Intelligence: Amazon RDS crea e salva automaticamente i backup delle istanze di database. Questi backup vengono conservati per un periodo di tempo specificato dall'utente. Assicurati di configurare la durata del backup e le impostazioni di conservazione di Amazon RDS in base ai requisiti di protezione dei dati. Per ulteriori informazioni, consulta la sezione Backup e ripristino di Amazon RDS.
- Server gestiti, server di amministrazione e server a livello Web: assicurati di configurare gli snapshot di Amazon EBS in base ai requisiti di protezione e conservazione dei dati.
- Storage condiviso: puoi gestire il backup e il ripristino dei file archiviati in Amazon EFS utilizzando AWS Backup. Il servizio AWS Backup può anche essere distribuito per gestire centralmente il backup e il ripristino di altri servizi, tra cui Amazon EC2, Amazon EBS e Amazon RDS. Per ulteriori informazioni, consulta Cos'è AWS Backup? Nella AWS Backup Developer Guide.

Sicurezza e conformità

Di seguito è riportato un elenco di best practice di sicurezza e servizi AWS che possono aiutarti a proteggere le tue applicazioni Oracle BI 12c nel cloud AWS:

- Crittografia a riposo: Amazon RDS, Amazon EFS e Amazon EBS supportano tutti algoritmi di crittografia standard del settore. Puoi utilizzare AWS Key Management Service (AWS KMS) per creare e gestire chiavi crittografiche e controllarne l'uso nei servizi AWS e nelle tue applicazioni. Puoi anche configurare Oracle Transparent Data Encryption (TDE) sull'istanza di database Amazon RDS for Oracle che ospita il tuo database di repository Oracle BI 12c.
- Crittografia in transito: è consigliabile attivare i protocolli SSL o TLS per proteggere i dati in transito tra i vari livelli dell'installazione di Oracle BI 12c. Puoi utilizzare AWS Certificate Manager (ACM) per fornire, gestire e distribuire certificati SSL e TLS pubblici e privati per le tue risorse Oracle BI 12c.
- Sicurezza di rete: assicurati di distribuire le tue risorse Oracle BI 12c in un Amazon VPC con i controlli di accesso appropriati configurati per il tuo caso d'uso. Configura i tuoi gruppi di sicurezza per filtrare il traffico in entrata e in uscita dalle EC2 istanze Amazon su cui è in esecuzione l'installazione. Inoltre, assicurati di configurare le liste di controllo degli accessi alla rete (NACLs) che consentono o negano il traffico in base a regole definite.
- Monitoraggio e registrazione: puoi usare AWS CloudTrail per tracciare le chiamate API alla tua infrastruttura AWS, incluse le risorse Oracle BI 12c. Questa funzionalità è utile per tenere traccia delle modifiche all'infrastruttura o per condurre un'analisi di sicurezza. Puoi anche utilizzare Amazon CloudWatch per visualizzare dati operativi che possono fornirti informazioni utili sulle prestazioni e sullo stato della tua applicazione Oracle BI 12c. Puoi configurare gli allarmi e

intraprendere azioni automatiche anche sulla base di tali allarmi. Amazon RDS fornisce strumenti di monitoraggio aggiuntivi, tra cui Enhanced Monitoring e Performance Insights.

Migrazione di uno stack ELK su Elastic Cloud su AWS

Battulga Purevragcha e Antony Prasad Thevaraj, Amazon Web Services

Buday Freddy, Nessuno

Riepilogo

Elastic fornisce servizi da molti anni, e i suoi utenti e clienti in genere gestiscono Elastic autonomamente in sede. Elastic Cloud , il servizio gestito di Elasticsearch, offre un modo per utilizzare Elastic Stack (ELK Stack) e soluzioni per la ricerca, l'osservabilità e la sicurezza aziendali. Puoi accedere alle soluzioni Elastic con app come Logs, Metrics, APM (monitoraggio delle prestazioni delle applicazioni) e SIEM (informazioni di sicurezza e gestione degli eventi). Puoi utilizzare funzionalità integrate come l'apprendimento automatico, la gestione del ciclo di vita degli indici, Kibana Lens (per le visualizzazioni drag-and-drop).

Quando passi da Elasticsearch autogestito a Elastic Cloud, il servizio Elasticsearch si occupa di quanto segue:

- Fornitura e gestione dell'infrastruttura sottostante
- Creazione e gestione di cluster Elasticsearch
- Scalabilità verso l'alto e verso il basso dei cluster
- Aggiornamenti, patch e acquisizione di istantanee

In questo modo avrai più tempo per concentrarti sulla risoluzione di altre sfide.

Questo modello definisce come migrare Elasticsearch 7.13 locale a Elasticsearch on Elastic Cloud on Amazon Web Services (AWS). Altre versioni potrebbero richiedere lievi modifiche ai processi descritti in questo modello. Per ulteriori informazioni, contatta il tuo rappresentante Elastic.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo con accesso ad Amazon Simple Storage Service (Amazon S3) per le istantanee
- Un collegamento privato sicuro e a larghezza di banda sufficientemente elevata per copiare i file di dati delle istantanee su Amazon S3

- Accelerazione dei trasferimenti Amazon S3
- Policy di Elastic Snapshot per garantire che l'inserimento dei dati venga archiviato regolarmente, in un data store locale sufficientemente grande o in uno storage remoto (Amazon S3)

È necessario comprendere le dimensioni delle istantanee e le politiche del ciclo di vita degli indici di accompagnamento in locale prima di iniziare la migrazione. Per ulteriori informazioni, contatta Elastic.

Ruoli e competenze

Il processo di migrazione richiede anche i ruoli e le competenze descritti nella tabella seguente.

Ruolo	Competenza	Responsabilità
Supporto per le app	Familiarità con Elastic Cloud ed Elastic on-premise	Tutte le attività relative a Elastic
Amministratore di sistema o DBA	Conoscenza approfondita dell'ambiente Elastic locale e della sua configurazione	La capacità di fornire storage, installare e utilizzare l'AWS Command Line Interface (AWS CLI) e identificare tutte le fonti di dati che alimentano Elastic in locale
Amministratore di rete	Conoscenza della connettiv ità, della sicurezza e delle prestazioni di rete on-premise e AWS	Creazione di collegamenti di rete dall'locale ad Amazon S3, con una comprensione della larghezza di banda della connettività

Limitazioni

- Elasticsearch on Elastic Cloud è disponibile solo nelle regioni AWS supportate (settembre 2021).

Versioni del prodotto

- Elasticsearch 7.13

Architecture

Stack tecnologico di origine

Elasticsearch 7.13 o versione successiva locale:

- Snapshot cluster
- Indicizza le istantanee
- Configurazione Beats

Architettura della tecnologia di origine

Il diagramma seguente mostra una tipica architettura locale con diversi metodi di ingestione, tipi di nodi e Kibana. I diversi tipi di nodi riflettono il cluster Elasticsearch, i ruoli di autenticazione e visualizzazione.

1. Ingestione da Beats a Logstash
2. Ingestione da Beats alla coda di messaggistica di Apache Kafka
3. Ingestione da Filebeat a Logstash
4. Inserimento dalla coda di messaggistica di Apache Kafka a Logstash
5. Ingestione da Logstash a un cluster Elasticsearch
6. Cluster Elasticsearch
7. Nodo di autenticazione e notifica
8. Nodi Kibana e blob

Stack tecnologico Target

Elastic Cloud viene distribuito sul tuo account SaaS (Software as a Service) in più regioni AWS con replica tra cluster.

- Snapshot cluster
- Indicizza le istantanee
- Configurazioni Beats
- Cloud elastico

- Network Load Balancer
- Amazon Route 53
- Simple Storage Service (Amazon S3)

Architettura Target

L'infrastruttura Elastic Cloud gestita è:

- Altamente disponibile, essendo presente in più zone di disponibilità e più regioni AWS.
- La regione è tollerante ai guasti perché i dati (indici e istantanee) vengono replicati utilizzando la replica tra cluster Elastic Cloud (CCR)
- Archiviazione, perché le istantanee vengono archiviate in Amazon S3
- Tolleranza alle partizioni di rete grazie a una combinazione di Network Load Balancer e Route 53
- Inserimento di dati provenienti da (ma non solo) Elastic APM, Beats, Logstash

Fasi di migrazione di alto livello

Elastic ha sviluppato una propria metodologia prescrittiva per la migrazione di Elastic Cluster on-premise a Elastic Cloud. La metodologia Elastic è direttamente allineata e complementare alle linee guida e alle best practice sulla migrazione di AWS, tra cui Well-Architected Framework e AWS Migration Acceleration Program (MAP). In genere, le tre fasi di migrazione AWS sono le seguenti:

- Valutazione
- Mobilitazione
- Migrazione e modernizzazione

Elastic segue fasi di migrazione simili con una terminologia complementare:

- Avviare
- Pianificazione
- Attuare
- Consegnare
- Chiudi

Elastic utilizza la metodologia Elastic Implementation per facilitare la consegna dei risultati del progetto. Ciò è stato progettato in modo inclusivo per garantire che Elastic, i team di consulenza e i team dei clienti collaborino con chiarezza per fornire congiuntamente i risultati attesi.

La metodologia Elastic combina la tradizionale fase a cascata con Scrum nella fase di implementazione. Le configurazioni dei requisiti tecnici vengono fornite in modo iterativo in modo collaborativo, riducendo al minimo i rischi.

Tools (Strumenti)

Servizi AWS

- Amazon Route 53 — Amazon Route 53 è un servizio Web DNS (Domain Name System) ad alta disponibilità e scalabilità. Puoi utilizzare Route 53 per eseguire tre funzioni principali in qualsiasi combinazione: registrazione dominio, routing DNS e controllo dell'integrità.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti. È possibile utilizzare Amazon S3 per memorizzare e recuperare qualsiasi volume di dati, in qualunque momento e da qualunque luogo tramite il Web. Questo modello utilizza un bucket S3 e Amazon S3 Transfer Acceleration.
- Elastic Load Balancing: Elastic Load Balancing distribuisce automaticamente il traffico in entrata su più destinazioni, EC2 come istanze, contenitori e indirizzi IP, in una o più zone di disponibilità.

Altri strumenti

- Beats: Beats invia dati da Logstash o Elasticsearch
- Elastic Cloud — Elastic Cloud è un servizio gestito per l'hosting di Elasticsearch.
- Elasticsearch: Elasticsearch è un motore di ricerca e analisi che utilizza Elastic Stack per archiviare centralmente i dati per ricerche e analisi su larga scala. Questo modello utilizza anche la creazione di istantanee e la replica tra cluster.
- Logstash: Logstash è una pipeline di elaborazione dati lato server che acquisisce dati da più fonti, li trasforma e quindi li invia all'archivio dati.

Epiche

Prepara la migrazione

Operazione	Description	Competenze richieste
Identifica i server che eseguono la soluzione Elastic locale.	Verifica che la migrazione elastica sia supportata.	Proprietario dell'app
Comprendi la configurazione del server locale.	Per comprendere la configurazione del server necessari a per gestire correttamente i carichi di lavoro in locale, individua l'ingombro hardware del server, la configurazione di rete e le caratteristiche di storage attualmente in uso	Supporto per app
Raccogli informazioni sull'account utente e sull'app.	Identifica i nomi utente e i nomi delle app utilizzati dall'ambiente Elastic locale.	Amministratore di sistema, supporto per le app
Configurazione di Document Beats e data shipper.	Per documentare le configurazioni, consulta le fonti di dati e i sink esistenti. Per ulteriori informazioni, consulta la documentazione di Elastic.	Supporto per le app
Determina la velocità e il volume dei dati.	Stabilisci una linea di base per la quantità di dati gestita dal cluster.	Amministratore di sistema, supporto per le app
Documenta gli scenari RPO e RTO.	Documenta gli scenari di Recovery Point Objective (RPO) e Recovery Time Objective (RTO) in termini di	Proprietario dell'app, amministratore di sistema, supporto dell'app

Operazione	Description	Competenze richieste
	interruzioni e accordi sui livelli di servizio (). SLAs	
Determina le impostazioni ottimali del ciclo di vita delle istantanee.	Definisci la frequenza con cui i dati devono essere protetti utilizzando istantane e elastiche durante e dopo la migrazione.	Proprietario dell'app, amministratore di sistema, supporto dell'app
Definisci le aspettative prestazionali dopo la migrazione.	Genera metriche sull'aggi ornamento attuale e previsto dello schermo, sui tempi di esecuzione delle query e sui comportamenti dell'interfaccia utente.	Amministratore di sistema, supporto per le app
Documenta i requisiti di accesso a Internet, trasporto, larghezza di banda e disponibi lità.	Verifica la velocità, la latenza e la resilienza delle connessio ni Internet per copiare le istantanee su Amazon S3.	Amministratore di rete
Documenta i costi correnti del runtime locale per Elastic.	Assicurati che il dimension amento dell'ambiente AWS di destinazione sia progettato per offrire prestazioni elevate e un ottimo rapporto qualità-prezzo.	DBA, amministratore di sistema, supporto per le app
Identifica le esigenze di autenticazione e autorizza zione.	Le funzionalità di sicurezza Elastic Stack forniscono funzionalità integrate come Lightweight Directory Access Protocol (LDAP), Security Assertion Markup Language (SAML) e OpenID Connect (OIDC).	DBA, amministratore di sistema, supporto delle app

Operazione	Description	Competenze richieste
Comprendi i requisiti normativi specifici in base alla posizione geografica.	Assicurati che i dati vengano esportati e crittografati in base ai tuoi requisiti e a qualsiasi requisito nazionale pertinente.	DBA, amministratore di sistema, supporto per le app

Implementa la migrazione

Operazione	Description	Competenze richieste
Prepara l'area di staging su Amazon S3.	Per ricevere istantanee su Amazon S3, crea un bucket S3 e un ruolo AWS Identity and Access Management (IAM) temporaneo con accesso completo al bucket appena creato. Per ulteriori informazioni, consulta Creazione di un ruolo per delegare le autorizzazioni a un utente IAM. Utilizza AWS Security Token Service per richiedere credenziali di sicurezza temporanee. Mantieni protetti l'ID della chiave di accesso, la chiave di accesso segreta e il token di sessione. Abilita Amazon S3 Transfer Acceleration nel bucket.	Amministratore AWS
Installa l'interfaccia a riga di comando di AWS e il plug-in Amazon S3 in locale.	Su ogni nodo Elasticsearch, esegui il comando seguente.	Amministratore AWS

Operazione	Description	Competenze richieste
	`sudo bin/elasticsearch-plugin install repository-s3` Quindi riavvia il nodo.	

Configura l'accesso al client Amazon S3.	Aggiungi le chiavi create in precedenza eseguendo i seguenti comandi.
	
	`elasticsearch-keystore add s3.client.default.access_key`
	
	`elasticsearch-keystore add s3.client.default.secret_key`
	
	`elasticsearch-keystore add s3.client.default.session_token`
		Amministratore AWS

Registra un repository di istantanee per dati elastici	Usa i Kibana Dev Tools per indicare al cluster locale locale locale su quale bucket S3 remoto scrivere.	Amministratore AWS

Operazione	Description	Competenze richieste
Configura la politica sulle istantanee.	Per configurare la gestione del ciclo di vita delle istantanee, nella scheda Kibana Policies, scegli la politica SLM e definisci quali orari, flussi di dati o indici devono essere inclusi e quali nomi usare. Configura una policy che acquisisca istantanee frequenti. Le istantanee sono incrementali e fanno un uso efficiente dello storage. Abbina la tua decisione di valutazio ne della prontezza. Una policy può anche specificare una policy di conservazione ed eliminare automaticamente le istantanee quando non sono più necessarie.	Supporto per le app
Verifica che le istantanee funzionino.	In Kibana Dev Tools, esegui il seguente comando. `GET _snapshot/<your_re po_name>/_all`	Amministratore AWS, supporto per app,
Implementa un nuovo cluster su Elastic Cloud.	Accedi a Elastic e scegli un cluster per «osservabilità, ricerca o sicurezza» basato sui risultati della tua attività nella valutazione della fattibili tà.	Amministratore AWS, supporto per app

Operazione	Description	Competenze richieste
Configura l'accesso al cluster key store.	Il nuovo cluster deve accedere al bucket S3 che memorizzerà le istantanee. In Elasticsearch Service Console, scegli Sicurezza e inserisci le chiavi di accesso e le chiavi IAM segrete che hai creato in precedenza.	Amministratore AWS
Configura il cluster ospitato su Elastic Cloud per accedere ad Amazon S3.	Configura un nuovo accesso del cluster all'archivio di snapshot creato in precedenza in Amazon S3. Usando Kibana, procedi come segue: 1. Scegli Stack Management, Snapshot Settings,. RegisterRepo 2. Nel campo Alias, inserisci il nome del repository. 3. Per il nome del client S3, scegli secondario. 4. Aggiungi il bucket S3 che hai creato in precedenza al repository. 5. Scegli Comprimi istantanea. 6. Per le impostazioni di crittografia, mantieni i valori predefiniti.	Amministratore AWS, App Support
Verifica il nuovo repository Amazon S3.	Assicurati di poter accedere al tuo nuovo repository ospitato nel cluster Elastic Cloud.	Amministratore AWS

Operazione	Description	Competenze richieste

Inizializza il cluster di servizi Elasticsearch.	Sulla Elasticsearch Service Console, inizializza il cluster di servizi Elasticsearch dalla snapshot S3.
	
	Esegui i seguenti comandi come POST.
	
	`*/_close?expand_wildcards=all`
	
	`/_snapshot/<your-repo-name>/ <your-snapshot-name>/_restore`
	
	`*/_open?expand_wildcards=all`
		Supporto per app

Completa la migrazione

Operazione	Description	Competenze richieste
Verificare che il ripristino dell'istantanea sia stato eseguito correttamente.	Usando Kibana Dev Tools, esegui il seguente comando. `GET _cat/indices`	Supporto per le app
Ridistribuire i servizi di ingestione.	Connect gli endpoint per Beats e Logstash al nuovo endpoint del servizio Elasticsearch.	Supporto per le app

Testa l'ambiente del cluster e pulisci

Operazione	Description	Competenze richieste
Convalida l'ambiente del cluster.	Dopo la migrazione dell'ambiente cluster elastico locale su AWS, puoi connetterti ad esso e utilizzare i tuoi strumenti di test di accettazione degli utenti (UAT) per convalidare il nuovo ambiente.	Supporto per le app
Pulisci le risorse.	Dopo aver verificato la corretta migrazione del cluster, rimuovi il bucket S3 e il ruolo IAM utilizzato per la migrazione.	Amministratore AWS

Risorse correlate

Riferimenti elastici

- Elastic Cloud
- Elasticsearch e Kibana gestiti su AWS
- Ricerca aziendale elastica
- Integrazioni elastiche
- Osservabilità elastica
- Sicurezza elastica
- Beats
- APM elastico
- Esegui la migrazione alla gestione del ciclo di vita degli indici
- Abbonamenti elastici
- Contatta Elastic

Post sul blog Elastic

- Come migrare da Elasticsearch autogestito a Elastic Cloud on AWS (post sul blog)
- Migrazione a Elastic Cloud (post sul blog)

Documentazione elastica

- Tutorial: automatizza i backup con SLM
- ILM: gestisci il ciclo di vita dell'indice
- Logstash
- Replica tra cluster (CCR)
- Acquisisci pipeline
- Esegui richieste API Elasticsearch
- Conservazione degli snapshot

Video e webinar elastici

- Migrazione flessibile al cloud
- Elastic Cloud: perché i clienti migrano (webinar)

Riferimenti AWS

- Elastic Cloud su AWS Marketplace
- Interfaccia a riga di comando di AWS
- AWS Direct Connect
- Programma di accelerazione della migrazione AWS
- Network Load Balancers
- Regioni e zone di disponibilità
- Amazon Route 53
- Amazon Simple Storage Service
- Accelerazione dei trasferimenti Amazon S3
- Connessioni VPN
- Well-Architected Framework

Informazioni aggiuntive

Se hai intenzione di migrare carichi di lavoro complessi, contatta Elastic Consulting Services. Se hai domande di base relative a configurazioni e servizi, contatta il team di Elastic Support.

Migrare i dati verso il Cloud AWS utilizzando Starburst

Antony Prasad Thevaraj e Suresh Veeragoni, Amazon Web Services

Shaun Van Staden, Nessuno

Riepilogo

Starburst aiuta ad accelerare il percorso di migrazione dei dati verso Amazon Web Services (AWS) fornendo un motore di query aziendale che riunisce le fonti di dati esistenti in un unico punto di accesso. Puoi eseguire analisi su più fonti di dati per ottenere informazioni preziose, prima di finalizzare qualsiasi piano di migrazione. Senza interrompere l' business-as-usualanalisi, puoi migrare i dati utilizzando il motore Starburst o un'applicazione dedicata di estrazione, trasformazione e caricamento (ETL).

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un cloud privato virtuale (VPC)
- Un cluster Amazon Elastic Kubernetes Service (Amazon EKS)
- Un gruppo Amazon Elastic Compute Cloud (Amazon EC2) Auto Scaling
- Un elenco degli attuali carichi di lavoro di sistema che devono essere migrati
- Connettività di rete dall' AWS ambiente locale

Architecture

Architettura di riferimento

Il seguente diagramma di architettura di alto livello mostra l'implementazione tipica di Starburst Enterprise in: Cloud AWS

1. Il cluster Starburst Enterprise funziona all'interno del tuo. Account AWS
2. Un utente si autentica utilizzando Lightweight Directory Access Protocol (LDAP) o Open Authorization (OAuth) e interagisce direttamente con il cluster Starburst.
3. Starburst può connettersi a diverse fonti di AWS dati, come AWS Glue Amazon Simple Storage Service (Amazon S3), Amazon Relational Database Service (Amazon RDS) e Amazon Redshift.

Starburst offre funzionalità di interrogazione federate su diverse fonti di dati in ambiente cloud Cloud AWS, on-premise o in altri ambienti cloud.

4. Puoi avviare Starburst Enterprise in un cluster Amazon EKS utilizzando i grafici Helm.
5. Starburst Enterprise utilizza i gruppi Amazon Auto EC2 Scaling e le istanze EC2 Amazon Spot per ottimizzare l'infrastruttura.
6. Starburst Enterprise si collega direttamente alle fonti di dati locali esistenti per leggere i dati in tempo reale. Inoltre, se disponi di un'implementazione Starburst Enterprise esistente in questo ambiente, puoi connettere direttamente il tuo nuovo cluster Starburst a questo cluster esistente. Cloud AWS

Tieni presente quanto segue:

- Starburst non è una piattaforma di virtualizzazione dei dati. È un motore di query MPP (Massively Parallel Processing) basato su SQL che costituisce la base di una strategia globale di data mesh per l'analisi.
- Quando Starburst viene distribuito come parte di una migrazione, dispone di una connettività diretta all'infrastruttura locale esistente.
- Starburst fornisce diversi connettori aziendali e open source integrati che facilitano la connettività a una varietà di sistemi legacy. Per un elenco completo dei connettori e delle relative funzionalità, consulta Connettori nella guida per l'utente di Starburst Enterprise.
- Starburst può interrogare i dati in tempo reale da fonti di dati locali. In questo modo si evitano interruzioni delle normali operazioni aziendali durante la migrazione dei dati.
- Se state migrando da un'implementazione Starburst Enterprise locale esistente, potete utilizzare un connettore speciale, Starburst Stargate, per connettere il cluster Starburst Enterprise direttamente al cluster locale. AWS Ciò offre ulteriori vantaggi in termini di prestazioni quando gli utenti aziendali e gli analisti di dati uniscono le query dall'ambiente locale all'ambiente locale. Cloud AWS

Panoramica dei processi di alto livello

Puoi accelerare i progetti di migrazione dei dati utilizzando Starburst perché Starburst consente di ottenere informazioni dettagliate su tutti i tuoi dati, prima della migrazione. L'immagine seguente mostra il processo tipico di migrazione dei dati utilizzando Starburst.

Ruoli

I seguenti ruoli sono in genere necessari per completare una migrazione utilizzando Starburst:

- Amministratore cloud: responsabile della disponibilità delle risorse cloud per l'esecuzione dell'applicazione Starburst Enterprise
- Amministratore Starburst: responsabile dell'installazione, della configurazione, della gestione e del supporto dell'applicazione Starburst
- Ingegnere dei dati — Responsabile di:
  - Migrazione dei dati legacy nel cloud
  - Creazione di viste semantiche per supportare l'analisi
- Proprietario della soluzione o del sistema: responsabile dell'implementazione complessiva della soluzione

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire Kubernetes su AWS Kubernetes senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.

Altri strumenti

- Helm: Helm è un gestore di pacchetti per Kubernetes che ti aiuta a installare e gestire le applicazioni sul tuo cluster Kubernetes.
- Starburst Enterprise — Starburst Enterprise è un motore di query MPP (Massively Parallel Processing) basato su SQL che costituisce la base di una strategia globale di data mesh per l'analisi.
- Starburst Stargate - Starburst Stargate collega cataloghi e fonti di dati in un ambiente Starburst Enterprise, come un cluster in un data center locale, ai cataloghi e alle fonti di dati in un altro ambiente Starburst Enterprise, come un cluster in Cloud AWS.

Epiche

Valuta i dati

Operazione	Description	Competenze richieste
Identifica e dai priorità ai tuoi dati.	Identifica i dati che desideri spostare. I sistemi legacy locali di grandi dimension i possono includere dati principali che desideri migrare insieme a dati che non desideri spostare o che non possono essere spostati per motivi di conformità. Iniziare con un inventario dei dati ti aiuta a stabilire la priorità dei dati a cui rivolgerti per primi. Per ulteriori informazi oni, consulta Introduzione alla scoperta automatica dei portafogli.	Ingegnere dei dati, DBA
Esplora, archivia ed esegui il backup dei tuoi dati.	Convalida la qualità, la quantità e la pertinenza dei dati per il tuo caso d'uso. Esegui il backup o crea un'istantanea dei dati secondo necessità e finalizza l'ambient e di destinazione per i dati.	Ingegnere dei dati, DBA

Configura l'ambiente Starburst Enterprise

Operazione	Description	Competenze richieste
Configura Starburst Enterprise in. Cloud AWS	Durante la catalogazione dei dati, configura Starburst	Amministratore AWS, sviluppatore di app

Operazione	Description	Competenze richieste
	Enterprise in un cluster Amazon EKS gestito. Per ulteriori informazioni, consulta Deploying with Kubernetes nella documentazione di riferimento di Starburst Enterprise. Ciò consente l' business-as-usualanalisi mentre è in corso la migrazione dei dati.	
Connect Starburst alle fonti di dati.	Dopo aver identificato i dati e configurato Starburst Enterprise, collega Starburst alle fonti di dati. Starburst legge i dati direttamente dalla fonte dati come una query SQL. Per ulteriori informazioni, consultate la documentazione di riferimento di Starburst Enterprise.	Amministratore AWS, sviluppatore di app

Migra i dati

Operazione	Description	Competenze richieste
Crea ed esegui le pipeline ETL.	Inizia il processo di migrazione dei dati. Questa attività può avvenire contemporaneamente all' business-as-usualanalisi. Per la migrazione, puoi utilizzare un prodotto di terze parti o Starburst. Starburst ha la capacità di leggere e scrivere dati da	Ingegnere dei dati

Operazione	Description	Competenze richieste
	fonti diverse. Per ulteriori informazioni, consultate la documentazione di riferimento di Starburst Enterprise.	
Convalida i dati.	Dopo la migrazione dei dati, convalida i dati per assicurarti che tutti i dati richiesti siano stati spostati e siano intatti.	Ingegnere dei dati, ingegnere DevOps

Tagliare

Operazione	Description	Competenze richieste
Taglia i dati.	Una volta completata la migrazione e la convalida dei dati, puoi tagliare i dati. Ciò comporta la modifica dei collegamenti di connessione dati in Starburst. Invece di puntare alle fonti locali, si punta alle nuove fonti cloud e si aggiornano le viste semantiche. Per ulteriori informazioni, vedete Connettori nella documentazione di riferimento di Starburst Enterprise.	Ingegnere dei dati, responsabile Cutover
Distribuiscilo agli utenti.	I consumatori di dati iniziano a utilizzare le fonti di dati migrate. Questo processo è invisibile agli utenti finali dell'analisi.	Responsabile Cutover, ingegnere dei dati

Risorse correlate

Marketplace AWS

- Galassia Starburst
- Starburst Enterprise
- Dati Starburst JumpStart
- Starburst Enterprise con Graviton

Documentazione Starburst

- Guida per l'utente di Starburst Enterprise
- Documentazione di riferimento di Starburst Enterprise

Altra documentazione AWS

- Inizia con l'individuazione automatica del portafoglio (AWS Prescriptive Guidance)
- Ottimizzazione dei costi e delle prestazioni dell'infrastruttura cloud con Starburst on (post sul blog) AWSAWS

Ottimizza l'ingestione ETL delle dimensioni dei file di input su AWS

Sostieni Patrikar, Amazon Web Services

Riepilogo

Questo modello mostra come ottimizzare la fase di inserimento del processo di estrazione, trasformazione e caricamento (ETL) per i carichi di lavoro Big Data e Apache Spark su AWS Glue ottimizzando le dimensioni dei file prima dell'elaborazione dei dati. Usa questo schema per prevenire o risolvere il problema dei file di piccole dimensioni. Cioè, quando un numero elevato di file di piccole dimensioni rallenta l'elaborazione dei dati a causa della dimensione aggregata dei file. Ad esempio, centinaia di file che pesano solo poche centinaia di kilobyte ciascuno possono rallentare in modo significativo la velocità di elaborazione dei dati per i tuoi job AWS Glue. Questo perché AWS Glue deve eseguire funzioni di elenco interne su Amazon Simple Storage Service (Amazon S3) e YARN (Yet Another Resource Negotiator) deve archiviare una grande quantità di metadati. Per migliorare la velocità di elaborazione dei dati, puoi utilizzare il raggruppamento per consentire alle attività ETL di leggere un gruppo di file di input in un'unica partizione in memoria. La partizione raggruppa automaticamente i file più piccoli. In alternativa, è possibile utilizzare codice personalizzato per aggiungere logica batch ai file esistenti.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Uno o più lavori AWS Glue
- Uno o più carichi di lavoro Big Data o Apache Spark
- Un bucket S3

Architecture

Lo schema seguente mostra come i dati in diversi formati vengono elaborati da un job AWS Glue e quindi archiviati in un bucket S3 per ottenere visibilità sulle prestazioni.

Il diagramma mostra il flusso di lavoro seguente:

1.
> **Note**
>
> Un job AWS Glue converte file di piccole dimensioni in formato CSV, JSON e Parquet in frame dinamici. : La dimensione del file di input ha l'impatto più significativo sulle prestazioni del job AWS Glue.

2. Il job AWS Glue esegue funzioni di elenco interne in un bucket S3.

Tools (Strumenti)

- AWS Glue è un servizio ETL completamente gestito. Ti aiuta a classificare, pulire, arricchire e spostare i dati in modo affidabile tra archivi e flussi di dati.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Epiche

Usa il raggruppamento per ottimizzare l'ingestione di ETL durante la lettura

Operazione	Description	Competenze richieste
Specificare la dimensione del gruppo.	Se hai più di 50.000 file, il raggruppamento viene eseguito per impostazione predefinita. Tuttavia, è possibile utilizzare il raggruppamento anche per meno di 50.000 file specificando la dimensione del gruppo nel parametro. `connectionOptions` Il `connectionOptions` parametro si trova nel metodo. `create_dynamic_frame.from_options`	Ingegnere dei dati

Operazione	Description	Competenze richieste
Scrivi il codice di raggruppamento.	Usa il `create_dynamic_frame` metodo per creare una cornice dinamica. Esempio: `S3bucket_node1 = glueContext.create_dynamic_frame.from_options(format_options={"multiline": False}, connection_type="s3", format="json", connection_options={ "paths": ["s3://bucket/prefix/file.json"], "recurse": True, "groupFiles": 'inPartition', "groupSize": 1048576 }, transformation_ctx="S3bucket_node1",)`  **Note** `groupFiles` Da utilizzare per raggruppare i file in un gruppo di partizioni Amazon S3. Si usa `groupSize`	Ingegnere dei dati

Operazione	Description	Competenze richieste
	per impostare la dimensione di destinazione del gruppo da leggere in memoria. Specificare `groupSize` in byte (1048576 = 1 MB).	
Aggiungi il codice al flusso di lavoro.	Aggiungi il codice di raggruppamento al tuo flusso di lavoro in AWS Glue.	Ingegnere dei dati

Utilizza una logica personalizzata per ottimizzare l'ingestione di ETL

Operazione	Description	Competenze richieste
Scegli la lingua e la piattaforma di elaborazione.	Scegli il linguaggio di scripting e la piattaforma di elaborazione su misura per il tuo caso d'uso.	Architetto del cloud
Scrivi il codice.	Scrivi la logica personalizzata per raggruppare i tuoi file.	Architetto del cloud
Aggiungi il codice al flusso di lavoro.	Aggiungi il codice al tuo flusso di lavoro in AWS Glue. Ciò consente di applicare la logica personalizzata ogni volta che il lavoro viene eseguito.	Ingegnere dei dati

Ripartizione durante la scrittura dei dati dopo la trasformazione

Operazione	Description	Competenze richieste
Analizza i modelli di consumo.	Scopri come le applicazioni downstream utilizzeranno i dati che scrivi. Ad esempio, se eseguono query sui dati ogni giorno e i dati vengono partizionati solo per regione o se i file di output sono molto piccoli, ad esempio 2,5 KB per file, non si tratta di una soluzione ottimale per il consumo.	DBA
Ripartiziona i dati prima della scrittura.	Ripartizione basata su join o interrogazioni durante l'elaborazione (in base alla logica di elaborazione) e dopo l'elaborazione (in base al consumo). Ad esempio, ripartizione basata sulla dimensione dei byte, ad esempio, o ripartizione basata su colonne `.repartition(100000)` , ad esempio. `.repartition("column_name")`	Ingegnere dei dati

Risorse correlate

- Lettura dei file di input in gruppi più grandi
- Monitoraggio di AWS Glue
- Monitoraggio di AWS Glue utilizzando i CloudWatch parametri di Amazon
- Monitoraggio e debug dei processi

- Guida introduttiva all'ETL serverless su AWS Glue

Informazioni aggiuntive

Determinazione della dimensione del

Non esiste un modo semplice per determinare se la dimensione di un file è troppo grande o troppo piccola. L'impatto della dimensione del file sulle prestazioni di elaborazione dipende dalla configurazione del cluster. In Hadoop di base, si consiglia di utilizzare file di 128 MB o 256 MB per sfruttare al meglio la dimensione del blocco.

Per la maggior parte dei carichi di lavoro di file di testo su AWS Glue, consigliamo una dimensione di file compresa tra 100 MB e 1 GB per un cluster da 5-10 DPU. Per determinare la dimensione ottimale dei file di input, monitora la sezione di preelaborazione del job AWS Glue, quindi controlla l'utilizzo della CPU e della memoria del job.

Considerazioni aggiuntive

Se le prestazioni nelle fasi iniziali dell'ETL rappresentano un ostacolo, prendete in considerazione la possibilità di raggruppare o unire i file di dati prima dell'elaborazione. Se hai il controllo completo sul processo di generazione dei file, può essere ancora più efficiente aggregare i punti dati sul sistema di origine stesso prima che i dati grezzi vengano inviati ad AWS.

Orchestra una pipeline ETL con convalida, trasformazione e partizionamento utilizzando AWS Step Functions

Sandip Gangapadhyay, Amazon Web Services

Riepilogo

Questo modello descrive come creare una pipeline di estrazione, trasformazione e caricamento (ETL) senza server per convalidare, trasformare, comprimere e partizionare un set di dati CSV di grandi dimensioni per l'ottimizzazione delle prestazioni e dei costi. La pipeline è orchestrata AWS Step Functions e include funzionalità di gestione degli errori, tentativi automatici e notifica agli utenti.

Quando un file CSV viene caricato in una cartella sorgente del bucket Amazon Simple Storage Service (Amazon S3), la pipeline ETL inizia a funzionare. La pipeline convalida il contenuto e lo schema del file CSV di origine, trasforma il file CSV in un formato Apache Parquet compresso, partiziona il set di dati per anno, mese e giorno e lo archivia in una cartella separata per l'elaborazione degli strumenti di analisi.

Il codice che automatizza questo pattern è disponibile su, nella pipeline ETL con repository. GitHub AWS Step Functions

Prerequisiti e limitazioni

Prerequisiti

- Account AWS Un attivo.
- AWS Command Line Interface (AWS CLI) installato e configurato con il tuo Account AWS, in modo da poter creare AWS risorse distribuendo uno AWS CloudFormation stack. Si consiglia di utilizzare la AWS CLI versione 2. Per istruzioni, consulta Installazione o aggiornamento alla versione più recente di AWS CLI nella AWS CLI documentazione. Per le istruzioni di configurazione, consulta Configurazione e impostazioni dei file di credenziali nella AWS CLI documentazione.
- Un bucket Amazon S3.
- Un set di dati CSV con lo schema corretto. (L'archivio di codice incluso in questo modello fornisce un file CSV di esempio con lo schema e il tipo di dati corretti che è possibile utilizzare.)
- Un browser Web che supporta il. Console di gestione AWS(Vedi l'elenco dei browser supportati).
- AWS Glue accesso alla console.
- AWS Step Functions accesso alla console.

Limitazioni

- In AWS Step Functions, il limite massimo per la conservazione dei registri cronologici è di 90 giorni. Per ulteriori informazioni, consulta le quote del servizio Step Functions nella AWS Step Functions documentazione.

Versioni del prodotto

- Python 3.13 per AWS Lambda
- AWS Glue versione 4.0

Architecture

Il flusso di lavoro illustrato nel diagramma è costituito da questi passaggi di alto livello:

1. L'utente carica un file CSV nella cartella di origine in Amazon S3.
2. Un evento di notifica di Amazon S3 avvia una AWS Lambda funzione che avvia la AWS Step Functions macchina a stati.
3. La funzione Lambda convalida lo schema e il tipo di dati del file CSV non elaborato.
4. A seconda dei risultati della convalida:
   a. Se la convalida del file sorgente ha esito positivo, il file viene spostato nella cartella dello stage per un'ulteriore elaborazione.
   b. Se la convalida fallisce, il file viene spostato nella cartella degli errori e viene inviata una notifica di errore tramite Amazon Simple Notification Service (Amazon SNS).
5. Un AWS Glue crawler crea lo schema del file raw dalla cartella stage in Amazon S3.
6. Un AWS Glue job trasforma, comprime e partiziona il file raw in formato Parquet.
7. Il AWS Glue processo sposta inoltre il file nella cartella Transform in Amazon S3.
8. Il AWS Glue crawler crea lo schema dal file trasformato. Lo schema risultante può essere utilizzato da qualsiasi processo di analisi. Puoi anche utilizzare Amazon Athena per eseguire query ad hoc.
9. Se la pipeline viene completata senza errori, il file dello schema viene spostato nella cartella di archivio. Se vengono rilevati errori, il file viene invece spostato nella cartella degli errori.
10. Amazon SNS invia una notifica che indica l'esito positivo o negativo in base allo stato di completamento della pipeline.

Tutte le AWS risorse utilizzate in questo modello sono serverless. Non ci sono server da gestire.

Tools (Strumenti)

Servizi AWS

- AWS Glue— AWS Glue è un servizio ETL completamente gestito che semplifica la preparazione e il caricamento dei dati per l'analisi da parte dei clienti.
- AWS Step Functions— AWS Step Functions è un servizio di orchestrazione senza server che consente di combinare AWS Lambda funzioni e altro Servizi AWS per creare applicazioni aziendali critiche. Attraverso la console AWS Step Functions grafica, puoi vedere il flusso di lavoro dell'applicazione come una serie di passaggi guidati dagli eventi.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti che offre scalabilità, disponibilità dei dati, sicurezza e prestazioni leader del settore.
- Amazon SNS — Amazon Simple Notification Service (Amazon SNS) è un servizio di messaggistica ad alta disponibilità, durevole, sicuro e completamente pub/sub gestito che consente di disaccoppiare microservizi, sistemi distribuiti e applicazioni serverless.
- AWS Lambda— AWS Lambda è un servizio di elaborazione che consente di eseguire codice senza fornire o gestire server. AWS Lambda esegue il codice solo quando necessario e ridimensiona automaticamente, da poche richieste al giorno a migliaia al secondo.

Codice

Il codice per questo pattern è disponibile su GitHub, nella pipeline ETL con repository. AWS Step Functions L'archivio del codice contiene i seguenti file e cartelle:

- `template.yml`— AWS CloudFormation modello per creare la pipeline ETL con. AWS Step Functions
- `parameter.json`— Contiene tutti i parametri e i valori dei parametri. Aggiorna questo file per modificare i valori dei parametri, come descritto nella sezione Epics.
- `myLayer/python`folder — Contiene i pacchetti Python necessari per creare il AWS Lambda layer richiesto per questo progetto.
- `lambda`folder — Contiene le seguenti funzioni Lambda:
  - `move_file.py`— Sposta il set di dati di origine nella cartella di archiviazione, trasformazione o errore.

- `check_crawler.py`— Controlla lo stato del AWS Glue crawler tante volte quante configurate dalla variabile di `RETRYLIMIT` ambiente prima di inviare un messaggio di errore.
- `start_crawler.py`— Avvia il crawler. AWS Glue
- `start_step_function.py`— Inizia. AWS Step Functions
- `start_codebuild.py`— Avvia il AWS CodeBuild progetto.
- `validation.py`— Convalida il set di dati grezzi di input.
- `s3object.py`— Crea la struttura di directory richiesta all'interno del bucket Amazon S3.
- `notification.py`— Invia notifiche di successo o di errore alla fine della pipeline.

Per utilizzare il codice di esempio, segui le istruzioni nella sezione Epics.

Epiche

Prepara i file sorgente

Operazione	Description	Competenze richieste
Clona il repository di codice di esempio.	1. Apri la pipeline ETL con il repository. AWS Step Functions 2. Scegliete Code nella pagina principale del repository, sopra l'elenco dei file, e copiate l'URL elencato in Clona con HTTPS. 3. Cambia la tua directory di lavoro nella posizione in cui desideri archiviare i file di esempio. 4. In un terminale o nel prompt dei comandi, digitate il comando: `git clone <repoURL>`	Developer

Operazione	Description	Competenze richieste
	dove `<repoURL>` si riferisce all'URL copiato nel passaggio 2.	

Operazione	Description	Competenze richieste
Aggiorna i valori dei parametri.	Nella copia locale del repository, modificate il `parameter.json` file e aggiornate i valori dei parametri predefiniti come segue: • `pS3BucketName` — Il nome del bucket Amazon S3 per l'archiviazione dei set di dati. Il modello creerà questo bucket per te. Il nome bucket deve essere univoco a livello globale. • `pSourceFolder` — Il nome della cartella all'interno del bucket Amazon S3 che verrà utilizzata per caricare il file CSV di origine. • `pStageFolder` — Il nome della cartella all'interno del bucket Amazon S3 che verrà utilizzata come area di gestione temporanea durante il processo. • `pTransformFolder` — Il nome della cartella all'interno del bucket Amazon S3 che verrà utilizzata per archiviare set di dati trasformati e partizionati. • `pErrorFolder` — La cartella all'interno del bucket Amazon S3 in cui verrà spostato il file CSV di	Developer

Operazione	Description	Competenze richieste
	origine se non può essere convalidato. • `pArchiveFolder` – Il nome della cartella all'interno del bucket Amazon S3 che verrà utilizzata per archiviare il file CSV di origine. • `pEmailforNotification` – Un indirizzo e-mail valido per ricevere notifiche. success/error • `pPrefix`– Una stringa di prefisso che verrà utilizzata nel nome del AWS Glue crawler. • `pDatasetSchema` – Lo schema del set di dati rispetto al quale verrà convalidato il file sorgente. Il pacchetto Cerberus Python viene utilizzato per la convalida del set di dati sorgente. Per ulteriori informazioni, consulta il sito Web di Cerberus.	

Operazione	Description	Competenze richieste
Carica il codice sorgente nel bucket Amazon S3.	Prima di distribuire il AWS CloudFormation modello che automatizza la pipeline ETL, devi impacchettare i file di origine per il modello e caricarli in un bucket Amazon S3. Per fare ciò, esegui il seguente comando con il tuo profilo preconfigurato: AWS CLI `aws cloudformation package --template-file template.yml --s3-bucket <bucket_name> --output-template-file packaged.template --profile <profile_name>` dove: • `<bucket_name>` è il nome di un bucket Amazon S3 esistente in Regione AWS cui desideri distribuire lo stack. Questo bucket viene utilizzato per archiviare il pacchetto di codice sorgente per il modello. CloudFormation • `<profile_name>` è un AWS CLI profilo valido che hai preconfigurato al	Developer

Operazione	Description	Competenze richieste
	momento della configura zione. AWS CLI	

Creazione dello stack

Operazione	Description	Competenze richieste
Implementa il CloudFormation modello.	Per distribuire il AWS CloudFormation modello, esegui il comando seguente: AWS CLI `aws cloudformation deploy --stack-name <stack_name> --template-file packaged.template --parameter-overrides file://parameter.json --capabilities CAPABILITY_IAM --profile <profile_name>` dove: • `<stack_name>` è un identificatore univoco per lo CloudFormation stack. • `<profile-name>` è il tuo profilo preconfigurato AWS CLI .	Developer
Controlla lo stato di avanzamento.	Sulla AWS CloudFormation console, controlla lo stato di avanzamento dello sviluppo dello stack. Quando lo stato	Developer

Operazione	Description	Competenze richieste
	è`CREATE_COMPLETE` , lo stack è stato distribuito correttamente.	
Annota il nome del AWS Glue database.	La scheda Output per lo stack mostra il nome del AWS Glue database. Il nome chiave è. `GlueDBOutput`	Developer

Prova la pipeline

Operazione	Description	Competenze richieste
Avvia la pipeline ETL.	1. Accedi alla cartella di origine (`source`o al nome della cartella che hai impostato nel `parameter.json` file) all'interno del bucket Amazon S3. 2. Carica un file CSV di esempio in questa cartella. (L'archivio del codice fornisce un file di esempio chiamato `Sample_Bank_Transaction_Raw_Dataset.csv` che puoi usare.) Il caricamento del file avvierà la pipeline ETL tramite Step Functions. 3. Nella console Step Functions, controlla lo stato della pipeline ETL.	Developer

Operazione	Description	Competenze richieste
Controlla il set di dati partizionato.	Al termine della pipeline ETL, verifica che il set di dati partizionato sia disponibile nella cartella di trasformazione di Amazon S3 (o nel nome della cartella che hai `transform` impostato nel file). `parameter.json`	Developer
Controlla il database partizionato. AWS Glue	1. Sulla AWS Glue console, seleziona il AWS Glue database creato dallo stack (questo è il database che hai annotato nell'epopea precedente). 2. Verifica che la tabella partizionata sia disponibile in. AWS Glue Data Catalog	Developer
Esegui interrogazioni.	(Facoltativo) Usa Amazon Athena per eseguire query ad hoc sul database partizionato e trasformato. Per istruzioni, consulta Esegui query SQL in Amazon Athena nella AWS documentazione.	Analista di database

risoluzione dei problemi

Problema	Soluzione
AWS Identity and Access Management autorizzazioni (IAM) per il AWS Glue job e il crawler	Se personalizzi ulteriormente il AWS Glue job o il crawler, assicurati di concedere le autorizzazioni IAM appropriate nel ruolo IAM utilizzato

Problema	Soluzione
	dal AWS Glue job o di fornire le autorizzazioni relative ai dati a. AWS Lake Formation Per ulteriori informazioni, consulta la documentazione relativa ad AWS.

Risorse correlate

Servizio AWS documentazione

- AWS Step Functions
- AWS Glue
- AWS Lambda
- Amazon S3
- Amazon SNS

Informazioni aggiuntive

Il diagramma seguente mostra il AWS Step Functions flusso di lavoro per una pipeline ETL di successo, dal pannello AWS Step Functions Inspector.

Il diagramma seguente mostra il AWS Step Functions flusso di lavoro per una pipeline ETL che fallisce a causa di un errore di convalida dell'input, dal pannello Step Functions Inspector.

Esegui analisi avanzate con Amazon Redshift ML

Po Hong e Chyanna Antonio, Amazon Web Services

Riepilogo

Sul cloud Amazon Web Services (AWS), puoi utilizzare l'apprendimento automatico di Amazon Redshift (Amazon Redshift ML) per eseguire analisi ML sui dati archiviati in un cluster Amazon Redshift o su Amazon Simple Storage Service (Amazon S3). Amazon Redshift ML supporta l'apprendimento supervisionato, che viene in genere utilizzato per analisi avanzate. I casi d'uso di Amazon Redshift ML includono la previsione dei ricavi, il rilevamento delle frodi con carte di credito e il Customer Lifetime Value (CLV) o le previsioni del tasso di abbandono dei clienti.

Amazon Redshift ML semplifica per gli utenti del database la creazione, il training e la distribuzione di modelli ML utilizzando comandi SQL standard. Amazon Redshift ML utilizza Amazon SageMaker Autopilot per addestrare e ottimizzare automaticamente i migliori modelli ML per la classificazione o la regressione in base ai dati, mantenendo il controllo e la visibilità.

Tutte le interazioni tra Amazon Redshift, Amazon S3 e SageMaker Amazon sono astratte e automatizzate. Una volta addestrato e distribuito, il modello ML diventa disponibile come funzione definita dall'utente (UDF) in Amazon Redshift e può essere utilizzato nelle query SQL.

Questo modello integra i modelli di creazione, addestramento e distribuzione di modelli ML in Amazon Redshift utilizzando SQL con Amazon Redshift ML dal blog AWS e il tutorial Build, train and deploy a ML SageMaker with Amazon dal Getting Started Resource Center.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Dati esistenti in una tabella Amazon Redshift

Competenze

- Familiarità con i termini e i concetti utilizzati da Amazon Redshift ML, tra cui apprendimento automatico, formazione e previsioni. Per ulteriori informazioni su questo argomento, consulta i modelli di Training ML nella documentazione di Amazon Machine Learning (Amazon ML).

- Esperienza con la configurazione degli utenti, la gestione degli accessi e la sintassi SQL standard di Amazon Redshift. Per ulteriori informazioni su questo argomento, consulta la sezione Guida introduttiva ad Amazon Redshift nella documentazione di Amazon Redshift.
- Conoscenza ed esperienza con Amazon S3 e AWS Identity and Access Management (IAM).
- Anche l'esperienza nell'esecuzione di comandi in AWS Command Line Interface (AWS CLI) è utile ma non obbligatoria.

Limitazioni

- Il cluster Amazon Redshift e il bucket S3 devono trovarsi nella stessa regione AWS.
- L'approccio di questo modello supporta solo modelli di apprendimento supervisionato come regressione, classificazione binaria e classificazione multiclasse.

Architecture

I passaggi seguenti spiegano come funziona Amazon Redshift ML SageMaker per creare, addestrare e distribuire un modello di machine learning:

1. Amazon Redshift esporta i dati di formazione in un bucket S3.
2. SageMaker Autopilot preelabora automaticamente i dati di allenamento.
3. Dopo aver richiamato l'`CREATE MODEL`istruzione, Amazon Redshift ML la SageMaker utilizza per la formazione.
4. SageMaker Autopilot cerca e consiglia l'algoritmo ML e gli iperparametri ottimali che ottimizzano le metriche di valutazione.
5. Amazon Redshift ML registra il modello ML di output come funzione SQL nel cluster Amazon Redshift.
6. La funzione del modello ML può essere utilizzata in un'istruzione SQL.

Stack tecnologico

- Amazon Redshift
- SageMaker
- Simple Storage Service (Amazon S3)

Tools (Strumenti)

- Amazon Redshift — Amazon Redshift è un servizio di data warehousing di livello aziendale, su scala petabyte, completamente gestito.
- Amazon Redshift ML — Amazon Redshift Machine Learning (Amazon Redshift ML) è un robusto servizio basato sul cloud che semplifica l'utilizzo della tecnologia ML da parte di analisti e data scientist di tutti i livelli.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è uno storage per Internet.
- Amazon SageMaker: SageMaker è un servizio di machine learning completamente gestito.
- Amazon SageMaker Autopilot — SageMaker Autopilot è un set di funzionalità che automatizza le attività chiave di un processo di apprendimento automatico (AutoML).

Codice

Puoi creare un modello di machine learning supervisionato in Amazon Redshift utilizzando il seguente codice:

```
"CREATE MODEL customer_churn_auto_model
FROM (SELECT state,
             account_length,
             area_code,
             total_charge/account_length AS average_daily_spend,
             cust_serv_calls/account_length AS average_daily_cases,
             churn
      FROM customer_activity
      WHERE  record_date < '2020-01-01'
     )
TARGET churn
FUNCTION ml_fn_customer_churn_auto
IAM_ROLE 'arn:aws:iam::XXXXXXXXXXXX:role/Redshift-ML'
SETTINGS (
  S3_BUCKET 'your-bucket'
);")
```

> **Note**
>
> Lo `SELECT` stato può fare riferimento alle tabelle normali di Amazon Redshift, alle tabelle esterne di Amazon Redshift Spectrum o a entrambe.

Epiche

Prepara un set di dati di addestramento e test

Operazione	Description	Competenze richieste
Preparare un set di dati di addestramento e test.	Accedi alla Console di gestione AWS e apri la SageMaker console Amazon. Segui le istruzioni del tutorial Build, train and deploy a machine learning model per creare un file.csv o Apache Parquet con una colonna di etichette (formazione supervisionata) e senza intestazione. **Note** Ti consigliamo di mescolare e suddividere il set di dati grezzi in un set di addestramento per l'addestramento del modello (70 percento) e un set di test per la valutazione delle prestazioni del modello (30 percento).	Data Scientist

Prepara e configura lo stack tecnologico

Operazione	Description	Competenze richieste
Crea e configura un cluster Amazon Redshift.	Sulla console Amazon Redshift, crea un cluster in base alle tue esigenze. Per ulteriori informazioni su questo argomento, consulta Creare un cluster nella documenta zione di Amazon Redshift. **Important** I cluster Amazon Redshift devono essere creati con il `SQL_PREVIEW` percorso di manutenzi one. Per ulteriori informazioni sulle tracce di anteprima, consulta Scelta delle tracce di manutenzi one del cluster nella documentazione di Amazon Redshift.	DBA, architetto del cloud
Crea un bucket S3 per archiviare i dati di allenamento e gli artefatti del modello.	Sulla console Amazon S3, crea un bucket S3 per i dati di addestramento e test. Per ulteriori informazioni sulla creazione di un bucket S3, consulta Creare un bucket S3 da AWS Quick Starts.	DBA, architetto cloud

Operazione	Description	Competenze richieste
	Important Assicurati che il cluster Amazon Redshift e il bucket S3 si trovino nella stessa regione.	
Crea e collega una policy IAM al cluster Amazon Redshift.	Crea una policy IAM per consentire al cluster Amazon Redshift di accedere SageMaker ad Amazon S3. Per istruzioni e passaggi, consulta Configurazione del cluster per l'utilizzo di Amazon Redshift ML nella documentazione di Amazon Redshift.	DBA, architetto cloud
Consenti a utenti e gruppi di Amazon Redshift di accedere a schemi e tabelle.	Concedi le autorizzazioni per consentire a utenti e gruppi in Amazon Redshift di accedere a schemi e tabelle interni ed esterni. Per passaggi e istruzioni, consulta Gestione delle autorizzazioni e della proprietà nella documentazione di Amazon Redshift.	DBA

Crea e addestra il modello ML in Amazon Redshift

Operazione	Description	Competenze richieste
Crea e addestra il modello ML in Amazon Redshift.	Crea e addestra il tuo modello di machine learning in Amazon Redshift ML. Per ulteriori	Sviluppatore, Data scientist

Operazione	Description	Competenze richieste
	informazioni, consulta la `CREATE MODEL` dichiarazione nella documentazione di Amazon Redshift.	

Esegui inferenze e previsioni in batch in Amazon Redshift

Operazione	Description	Competenze richieste
Esegui l'inferenza utilizzando la funzione del modello ML generata.	Per ulteriori informazioni sull'esecuzione dell'inferenza utilizzando la funzione del modello ML generata, consulta Prediction nella documentazione di Amazon Redshift.	Data scientist, utente di business intelligence

Risorse correlate

Preparare un set di dati di formazione e test

- Creazione, formazione e implementazione di un modello di machine learning con Amazon SageMaker

Prepara e configura lo stack tecnologico

- Creazione di un cluster Amazon Redshift
- Scelta dei percorsi di manutenzione dei cluster Amazon Redshift
- Creazione di un bucket S3
- Configurazione di un cluster Amazon Redshift per l'utilizzo di Amazon Redshift ML
- Gestione delle autorizzazioni e della proprietà in Amazon Redshift

Crea e addestra il modello ML in Amazon Redshift

- Istruzione CREATE MODEL in Amazon Redshift

Esegui inferenze e previsioni in batch in Amazon Redshift

- Previsione in Amazon Redshift

Altre risorse

- Guida introduttiva ad Amazon Redshift ML
- Creazione, addestramento e distribuzione di modelli ML in Amazon Redshift utilizzando SQL con Amazon Redshift ML
- Partner Amazon Redshift
- Partner per le competenze di apprendimento automatico di AWS

Interroga le tabelle Amazon DynamoDB con SQL utilizzando Amazon Athena

Gavin Perrie, Ajit Ambike e Brad Yates, Amazon Web Services

Riepilogo

Se i tuoi dati includono fonti diverse da Amazon Simple Storage Service (Amazon S3), puoi utilizzare query federate per accedere a tali origini dati relazionali, non relazionali, a oggetti o personalizzate. Questo modello mostra come configurare l'accesso alle query federate tramite Amazon Athena ad Amazon DynamoDB utilizzando un connettore di origine dati SQL.

Utilizzando questo pattern puoi fare quanto segue:

- Interroga le tabelle DynamoDB utilizzando SQL.
- Esegui query SQL federate in Athena e unisci tabelle DynamoDB con altre fonti di dati supportate.

Prerequisiti e limitazioni

Prerequisiti

- Una tabella DynamoDB.
- Un gruppo di lavoro Athena impostato per utilizzare il motore Athena versione 2. Per istruzioni, consulta la documentazione di Athena.
- Un bucket S3 in cui la `AthenaDynamoDBConnector` AWS Lambda funzione può trasferire i dati. Il bucket S3 e la funzione Lambda devono trovarsi nella stessa regione AWS.

Se è la prima volta che accedi ad Athena, avrai bisogno di un bucket S3 aggiuntivo da utilizzare come posizione dei risultati della query. Per istruzioni, consulta la documentazione di Athena.

Limitazioni

- Le operazioni di scrittura come INSERT INTO non sono supportate.

Versioni del prodotto

- Athena Query Federation rilascia il GitHub

Architecture

Architettura Target

Il diagramma seguente mostra il flusso di connessione dopo aver stabilito lo schema. L'utente si connette ad Amazon Athena per fornire la query. Athena passa la query e la destinazione alla funzione Lambda del connettore di origine dati DynamoDB, che recupera e restituisce i dati ad Athena. Se vengono restituite grandi quantità di dati, Athena archivia i risultati temporanei nel secchio di raccolta prima di imballare e restituire il set di dati completo.

Tools (Strumenti)

Servizi AWS

- Amazon Athena è un servizio di query interattivo che ti aiuta ad analizzare i dati direttamente in Amazon Simple Storage Service (Amazon S3) utilizzando SQL standard. Questo modello utilizza Amazon Athena DynamoDB Connector, uno strumento creato utilizzando Amazon Athena Query Federation SDK e installato come applicazione tramite. AWS Lambda AWS Serverless Application Repository
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub Athena Query Federation.

Epiche

Configurare e testare il connettore di origine dati DynamoDB

Operazione	Description	Competenze richieste
Implementa l' AthenaDynamoDBConnector applicazione.	Per eseguire la distribuzione AthenaDynamoDBConnecter, effettuate le seguenti operazioni: 1. Accedi a e scegli Regione AWS quello che stai utilizzando per la tabella DynamoDB e lo spill bucket. Console di gestione AWS 2. Aprire il Serverless https://console.aws.amazon.com/serverlessrepo/ Application Repository all'indirizzo. 3. Nel riquadro di navigazione, scegli Applicazioni disponibili. 4. Per l'accesso AWS Identity and Access Management (IAM), nella barra di ricerca, seleziona la casella di controllo Mostra app che creano ruoli IAM personalizzati o politiche delle risorse. 5. Cerca e seleziona AthenaDynamoDBConnectore assicurati che l'autore elencato sia Amazon Athena Federation.	AWS DevOps

Operazione	Description	Competenze richieste
	6. In Impostazioni dell'appl icazione, inserisci i seguenti valori: • SpillBucket– Luogo in cui la funzione può trasferire dati. • AthenaCatalogName– Il nome della funzione Lambda che verrà creata. Il nome verrà utilizzat o anche come nome dell'origine dati in Athena. 7. Seleziona la casella di controllo per confermare la creazione di ruoli e policy IAM. 8. Seleziona Implementa.	

Operazione	Description	Competenze richieste
Crea una fonte di dati per Athena.	Per creare l'origine dati, procedi come segue: 1. Apri la console Athena all'indirizzo https://console.aws.amazon.com/athena/. 2. Espandi il riquadro di navigazione e scegli Fonti di dati. 3. Seleziona Create data source (Crea origine dati). 4. Scegli Amazon DynamoDB. 5. Inserisci il nome della fonte di dati. 6. Seleziona la funzione Lambda che hai creato. 7. Esamina i dettagli e scegli Crea origine dati.	AWS DevOps

Operazione	Description	Competenze richieste
Usa Athena per interrogare la tabella DynamoDB.	Per interrogare la tabella DynamoDB, procedi come segue: 1. Nella console Athena, espandi il riquadro di navigazione e scegli Query editor. 2. Nell'elenco a discesa Origine dati, scegli l'origine dati che hai creato. 3. Verificare che le tabelle DynamoDB siano elencate in Tabelle. 4. Eseguire la query.	Sviluppatore di app

risoluzione dei problemi

Problema	Soluzione
La query ha esito negativo con. *`GENERIC_INTERNAL_ERROR: The bucket is in this region: <region>`*	Assicurati che lo spill bucket Athena e la funzione Lambda siano creati nello stesso ambiente. Regione AWS
L'origine dati appena creata non è visibile sulla console Athena.	I cataloghi di dati Athena sono regionali. Assicurati che sia `AthenaDynamoDBConnector` stato schierato nella regione in cui stai cercando di usare Athena.
Non è possibile eseguire la query sull'origine dati appena creata.	Verifica che la posizione dei risultati della query sia stata impostata.

Risorse correlate

- Connettore Amazon Athena DynamoDB
- Query federata Amazon Athena

Accedi, esegui query e unisciti a tabelle Amazon DynamoDB utilizzando Athena

Moinul Al-Mamun, Amazon Web Services

Riepilogo

Questo modello mostra come configurare una connessione tra Amazon Athena e Amazon DynamoDB utilizzando il connettore Amazon Athena DynamoDB. Il connettore utilizza una funzione AWS Lambda per interrogare i dati in DynamoDB. Non è necessario scrivere alcun codice per configurare la connessione. Una volta stabilita la connessione, è possibile accedere e analizzare rapidamente le tabelle DynamoDB utilizzando Athena Federated Query per eseguire comandi SQL da Athena. Puoi anche unire una o più tabelle DynamoDB tra loro o con altre fonti di dati, come Amazon Redshift o Amazon Aurora.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo con autorizzazioni per gestire tabelle DynamoDB, sorgenti di dati Athena, Lambda e ruoli AWS Identity and Access Management (IAM)
- Un bucket Amazon Simple Storage Service (Amazon S3) in cui Athena può archiviare i risultati delle query
- Un bucket S3 in cui il connettore Athena DynamoDB può salvare i dati a breve termine
- Una regione AWS che supporta la versione 2 del motore Athena
- Autorizzazioni IAM per accedere ad Athena e ai bucket S3 richiesti
- Connettore Amazon Athena DynamoDB, installato

Limitazioni

L'interrogazione delle tabelle DynamoDB comporta un costo. Le dimensioni delle tabelle che superano alcuni gigabyte (GBs) possono comportare costi elevati. Si consiglia di considerare i costi prima di eseguire qualsiasi operazione SCAN completa della tabella. Per ulteriori informazioni, consulta Prezzi di Amazon DynamoDB. Per ridurre i costi e ottenere prestazioni elevate, si consiglia di utilizzare sempre LIMIT nella query (ad esempio,`SELECT * FROM table1 LIMIT 10`). Inoltre, prima di eseguire una query JOIN o GROUP BY in un ambiente di produzione, considerate le

dimensioni delle tabelle. Se le tue tabelle sono troppo grandi, prendi in considerazione opzioni alternative come la migrazione della tabella su Amazon S3.

Architecture

Il diagramma seguente mostra come un utente può eseguire una query SQL su una tabella DynamoDB da Athena.

Il diagramma mostra il flusso di lavoro seguente:

1. Per interrogare una tabella DynamoDB, un utente esegue una query SQL da Athena.
2. Athena avvia una funzione Lambda.
3. La funzione Lambda interroga i dati richiesti nella tabella DynamoDB.
4. DynamoDB restituisce i dati richiesti alla funzione Lambda. Quindi, la funzione trasferisce i risultati della query all'utente tramite Athena.
5. La funzione Lambda memorizza i dati nel bucket S3.

Stack tecnologico

- Amazon Athena
- Amazon DynamoDB
- Simple Storage Service (Amazon S3)
- AWS Lambda

Tools (Strumenti)

- Amazon Athena è un servizio di query interattivo che ti aiuta ad analizzare i dati direttamente in Amazon S3 utilizzando SQL standard.
- Amazon Athena DynamoDB Connector è uno strumento AWS che consente ad Athena di connettersi a DynamoDB e accedere alle tabelle utilizzando query SQL.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.

Epiche

Creare tabelle DynamoDB di esempio

Operazione	Description	Competenze richieste
Crea la prima tabella di esempio.	1. Accedi alla Console di gestione AWS e apri la console DynamoDB. 2. Scegliere Create table (Crea tabella). 3. Per il nome della tabella, inserisci dydbtable1. 4. Per la chiave di partizione, immettere. PK1 5. Per Chiave di ordinamento, immetti SK1. 6. Nella sezione Impostazioni tabella, scegli Personalizza impostazioni. 7. Nella sezione Table class, scegli DynamoDB Standard. 8. Nella sezione Impostazioni della capacità di lettura/scrittura, per la modalità Capacità, scegli On-demand. 9. Nella sezione Encryption at rest, scegli Owned by Amazon DynamoDB. 10. Scegliere Create table (Crea tabella).	Developer

Operazione	Description	Competenze richieste

Inserisci dati di esempio nella prima tabella.	1. Aprire la console DynamoDB.
	2. Nel riquadro di navigazione, scegli Tabella, quindi scegli la tua tabella nella colonna Nome.
	3. Scegli Azioni, quindi scegli Crea elemento.
	4. Scegli la visualizzazione JSON.
	5. Nella barra del titolo dell'editor Attributi, disattiva View DynamoDB JSON.
	6. Nell'editor Attributi, inserisci i seguenti dati di esempio uno per uno:
	
	`{ "PK1": "1234", "SK1": "info", "Salary": "5000" }`
	
	`{ "PK1": "1235", "SK1": "info", "Salary": "5200" }`
		Developer

Operazione	Description	Competenze richieste
Create la seconda tabella di esempio.	1. Aprire la console DynamoDB. 2. Scegliere Create table (Crea tabella). 3. Per Nome tabella, immettete dydbtable2. 4. In Partition key (Chiave di partizione), inserisci PK2. 5. Per Chiave di ordinamento, immetti SK2. 6. Nella sezione Impostazioni tabella, scegli Personalizza impostazioni. 7. Nella sezione Table class, scegli DynamoDB Standard. 8. Nella sezione Impostazioni della capacità di lettura/scrittura, per la modalità Capacità, scegli On-demand. 9. Nella sezione Encryption at rest, scegli Owned by Amazon DynamoDB. 10. Scegliere Create table (Crea tabella).	Developer

Operazione	Description	Competenze richieste

Inserisci dati di esempio nella seconda tabella.	1. Aprire la console DynamoDB.
	2. Nel riquadro di navigazione, scegli Tabella, quindi scegli la tua tabella nella colonna Nome.
	3. Scegli Azioni, quindi scegli Crea elemento.
	4. Nella barra del titolo dell'editor Attributi, disattiva View DynamoDB JSON.
	5. Nell'editor Attributi, inserisci i seguenti dati di esempio uno per uno:
	`{ "PK2": "1234", "SK2": "bonus", "Bonus": "500" }`
	`{ "PK2": "1235", "SK2": "bonus", "Bonus": "1000" }`
		Developer

Creare un'origine dati in Athena per DynamoDB

Operazione	Description	Competenze richieste
Configura il connettore di origine dati.	Crea un'origine dati per DynamoDB, quindi crea	Developer

Operazione	Description	Competenze richieste
	una funzione Lambda per connetterti a quell'origine dati. 1. Accedi alla Console di gestione AWS e apri la console Athena. 2. Nel pannello di navigazione, scegli Origini dati, quindi scegli Crea origine dati. 3. Scegli l'origine dati Amazon DynamoDB, quindi scegli Avanti. 4. Nella sezione Dettagli dell'origine dati, per Nome dell'origine dati, inserisci TestDynamoDB. 5. Nella sezione Dettagli di connessione, seleziona una funzione Lambda già distribuita o scegli Crea funzione Lambda se non disponi di una funzione Lambda da utilizzare per questo modello. Nota: per ulteriori informazioni sulla creazione di una funzione Lambda, consulta Guida introduttiva a Lambda nella Lambda Developer Guide. 6. (Facoltativo) Se scegli la funzione Create Lambda, devi configurare il CloudFormation modello AWS incluso nell'appl	

Operazione	Description	Competenze richieste
	icazione Java prima di distribuire quello stack. Il modello include ApplicationName, SpillBucket AthenaCatalogName, e altre impostazioni dell'applicazione. Nota: dopo aver distribuito questa applicazione basata su Java, lo stack crea una funzione Lambda che consente ad Athena di comunicare con DynamoDB. In questo modo le tabelle sono accessibili tramite comandi SQL. 7. Implementa la tua funzione Lambda. 8. Scegli Next (Successivo).	

Operazione	Description	Competenze richieste
Verifica che la funzione Lambda possa accedere allo spill bucket S3.	1. Aprire la console Lambda. 2. Nel pannello di navigazione, scegli Funzioni, quindi scegli la funzione che hai creato in precedenza. 3. Scegli la scheda Configurazione. 4. Nel riquadro di sinistra, scegli Variabili di ambiente, quindi conferma che il valore della chiave `èspill_bucket` . 5. Nel riquadro di sinistra, scegli Autorizzazioni, quindi nella sezione Ruolo di esecuzione, scegli il ruolo IAM associato. Nota: vieni indirizzato al ruolo IAM collegato alla tua funzione Lambda nella console IAM. 6. Conferma di avere l'autorizzazione di scrittura su `spill_bucket` bucket. Se riscontri errori, consulta la sezione Informazioni aggiuntive di questo schema come guida.	Developer

Accedi alle tabelle DynamoDB da Athena

Operazione	Description	Competenze richieste
Interroga le tabelle DynamoDB.	1. Accedi alla Console di gestione AWS e apri la console Athena. 2. Nel pannello di navigazione, scegli Origini dati, quindi scegli Crea origine dati. 3. Nel riquadro di navigazione, scegli Query Editor (Editor della query). 4. Nella scheda Editor, nella sezione Dati, per Origine dati, scegli la tua fonte di dati per Origine dati. 5. Per Database, scegli il database. 6. Per Query 1, inserisci la seguente query: `SELECT * FROM dydbtable1 t1;` 7. Scegliete Esegui, quindi verificate l'output nella tabella. 8. Per Query 2, inserisci la seguente query: `SELECT * FROM dydbtable2 t2;` 9. Scegliete Esegui, quindi verificate l'output nella tabella.	Developer
Unisci le due tabelle DynamoDB.	DynamoDB è un data store NoSQL e non supporta l'operazione di join SQL. Di	Developer

Operazione	Description	Competenze richieste
	conseguenza, è necessario eseguire un'operazione di join su due tabelle DynamoDB: 1. Scegliete l'icona con il segno più per creare un'altra query. 2. Per Query 3, inserisci la seguente query: `SELECT pk1, salary, bonus FROM dydbtable1 t1 JOIN dydbtable2 t2 ON t1.pk1 = t2.pk2;`	

Risorse correlate

- Connettore Amazon Athena DynamoDB (AWS Labs)
- Interroga qualsiasi fonte di dati con la nuova query federata di Amazon Athena (AWS Big Data Blog)
- Riferimento alla versione del motore Athena (Athena User Guide)
- Semplifica l'estrazione e l'analisi dei dati di Amazon DynamoDB utilizzando AWS Glue e Amazon Athena (AWS Database Blog)

Informazioni aggiuntive

Se esegui una query in Athena con `spill_bucket` il `{bucket_name}/folder_name/` formato, puoi ricevere il seguente messaggio di errore:

```
"GENERIC_USER_ERROR: Encountered an exception[java.lang.RuntimeException] from your
 LambdaFunction[arn:aws:lambda:us-east-1:xxxxxx:function:testdynamodb] executed in
 context[retrieving meta-data] with message[You do NOT own the spill bucket with the
 name: s3://amzn-s3-demo-bucket/athena_dynamodb_spill_data/]
```

```
This query ran against the "default" database, unless qualified by the query. Please
 post the error message on our forum  or contact customer support with Query Id:
 [query-id]"
```

Per risolvere questo errore, aggiorna la variabile di ambiente della funzione Lambda `spill_bucket` a`{bucket_name_only}`, quindi aggiorna la seguente policy Lambda IAM per l'accesso in scrittura ai bucket:

```
{
             "Action": [
                 "s3:GetObject",
                 "s3:ListBucket",
                 "s3:GetBucketLocation",
                 "s3:GetObjectVersion",
                 "s3:PutObject",
                 "s3:PutObjectAcl",
                 "s3:GetLifecycleConfiguration",
                 "s3:PutLifecycleConfiguration",
                 "s3:DeleteObject"
             ],
             "Resource": [
                 "arn:aws:s3:::spill_bucket",
                 "arn:aws:s3:::spill_bucket/*"
             ],
             "Effect": "Allow"
         }
```

In alternativa, puoi rimuovere il connettore sorgente dati Athena creato in precedenza e ricrearlo utilizzando only for. `{bucket_name} spill_bucket`

Imposta uno spazio dati minimo praticabile per condividere i dati tra le organizzazioni

Ramy Hcini, Michael Miller e Jorge Hernández Suarez, Amazon Web Services

Ismail Abdellaoui e Malte Gasseling, Think-it

Riepilogo

Gli spazi dati sono reti federate per lo scambio di dati con fiducia e controllo dei propri dati come principi fondamentali. Consentono alle organizzazioni di condividere, scambiare e collaborare sui dati su larga scala offrendo una soluzione economica e indipendente dalla tecnologia.

Gli spazi di dati hanno il potenziale per promuovere in modo significativo gli sforzi per un futuro sostenibile utilizzando la risoluzione dei problemi basata sui dati con un end-to-end approccio che coinvolga tutte le parti interessate.

Questo modello illustra l'esempio di come due aziende possono utilizzare la tecnologia dello spazio dati su Amazon Web Services (AWS) per portare avanti la propria strategia di riduzione delle emissioni di carbonio. In questo scenario, la società X fornisce dati sulle emissioni di carbonio, che la società Y consuma. Consulta la sezione Informazioni aggiuntive per i seguenti dettagli sulle specifiche dello spazio dati:

- Partecipanti
- Caso aziendale
- Autorità per lo spazio dei dati
- Componenti dello spazio dati
- Servizi di spazio dati
- Dati da scambiare
- Modello di dati
- Connettore Tractus-X EDC

Il modello include passaggi per quanto segue:

- Implementazione dell'infrastruttura necessaria per uno spazio dati di base con due partecipanti in esecuzione. AWS
- Scambio di dati sulle emissioni di carbonio e sull'intensità utilizzando i connettori in modo sicuro.

Questo modello implementa un cluster Kubernetes che ospiterà i connettori dello spazio dati e i relativi servizi tramite Amazon Elastic Kubernetes Service (Amazon EKS).

Il piano di controllo e il piano dati di Eclipse Dataspace Components (EDC) sono entrambi implementati su Amazon EKS. Il grafico ufficiale Tractus-X Helm implementa i servizi PostgreSQL e Vault come dipendenze. HashiCorp

Inoltre, il servizio di identità viene distribuito su Amazon Elastic Compute Cloud EC2 (Amazon) per replicare uno scenario reale di uno spazio dati minimo valido (MVDS).

Prerequisiti e limitazioni

Prerequisiti

- Un utente attivo Account AWS per implementare l'infrastruttura nel paese prescelto Regione AWS
- Un utente AWS Identity and Access Management (IAM) con accesso ad Amazon S3 che verrà utilizzato temporaneamente come utente tecnico (il connettore EDC attualmente non supporta l'uso dei ruoli. Ti consigliamo di creare un utente IAM specifico per questa demo e che a tale utente siano associate autorizzazioni limitate.)
- AWS Command Line Interface (AWS CLI) installato e configurato nella versione prescelta Regione AWS
- AWS credenziali di sicurezza
- eksctl sulla tua postazione di lavoro
- Git sulla tua workstation
- kubectl
- timone
- Postino
- Un certificato AWS Certificate Manager (ACM) SSL/TLS
- Un nome DNS che punterà a un Application Load Balancer (il nome DNS deve essere coperto dal certificato ACM)
- HashiCorp Vault (per informazioni sull'utilizzo Gestione dei segreti AWS per la gestione dei segreti, consulta la sezione Informazioni aggiuntive).

Versioni del prodotto

- AWS CLI versione 2+

- Collezione Postman v2.1

Limitazioni

- Selezione del connettore – Questa distribuzione utilizza un connettore basato su EDC. Tuttavia, assicurati di considerare i punti di forza e le funzionalità dei connettori EDC e FIWARE True per prendere una decisione informata che sia in linea con le esigenze specifiche dell'implementazione.
- Costruzione del connettore EDC – La soluzione di implementazione scelta si basa sulla tabella Tractus-X EDC Connector Helm, un'opzione di implementazione consolidata e ampiamente testata. La decisione di utilizzare questo grafico è determinata dal suo utilizzo comune e dall'inclusione di estensioni essenziali nella build fornita. Sebbene PostgreSQL HashiCorp e Vault siano componenti predefiniti, hai la flessibilità di personalizzare la configurazione del tuo connettore, se necessario.
- Accesso privato al cluster – L'accesso al cluster EKS distribuito è limitato ai canali privati. L'interazione con il cluster viene eseguita esclusivamente tramite l'uso di `kubectl` un IAM. L'accesso pubblico alle risorse del cluster può essere abilitato utilizzando sistemi di bilanciamento del carico e nomi di dominio, che devono essere implementati in modo selettivo per esporre servizi specifici a una rete più ampia. Tuttavia, non è consigliabile fornire l'accesso pubblico.
- Attenzione alla sicurezza – L'accento è posto sull'astrazione delle configurazioni di sicurezza secondo le specifiche predefinite in modo da potersi concentrare sulle fasi relative allo scambio di dati dei connettori EDC. Sebbene vengano mantenute le impostazioni di sicurezza predefinite, è fondamentale abilitare comunicazioni sicure prima di esporre il cluster alla rete pubblica. Questa precauzione garantisce una solida base per la gestione sicura dei dati.
- Costo dell'infrastruttura – È possibile ottenere una stima del costo dell'infrastruttura utilizzando il. Calcolatore dei prezzi AWS Un semplice calcolo mostra che i costi possono arrivare fino a 162,92 USD al mese per l'infrastruttura implementata.

Architecture

L'architettura MVDS comprende due cloud privati virtuali (VPCs), uno per il servizio di identità Dynamic Attribute Provisioning System (DAPS) e uno per Amazon EKS.

Architettura DAPS

Il diagramma seguente mostra DAPS in esecuzione su EC2 istanze controllate da un gruppo Auto Scaling. Un Application Load Balancer e una tabella di routing espongono i server DAPS. Amazon Elastic File System (Amazon EFS) sincronizza i dati tra le istanze DAPS.

Architettura Amazon EKS

Gli spazi dati sono progettati per essere soluzioni indipendenti dalla tecnologia ed esistono diverse implementazioni. Questo modello utilizza un cluster Amazon EKS per distribuire i componenti tecnici dello spazio dati. Il diagramma seguente mostra l'implementazione del cluster EKS. I nodi di lavoro vengono installati in sottoreti private. I pod Kubernetes accedono all'istanza Amazon Relational Database Service (Amazon RDS) per PostgreSQL che si trova anche nelle sottoreti private. I pod Kubernetes archiviano i dati condivisi in Amazon S3.

Tools (Strumenti)

AWS servizi

- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita in tutte Account AWS le regioni.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon Elastic File System (Amazon EFS) ti aiuta a creare e configurare file system condivisi nel Cloud AWS.
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire AWS Kubernetes senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Elastic Load Balancing (ELB) distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni. Ad esempio, puoi distribuire il traffico tra EC2 istanze, contenitori e indirizzi IP in una o più zone di disponibilità.

Altri strumenti

- eksctl è un'utilità da riga di comando per la creazione e la gestione di cluster Kubernetes su Amazon EKS.
- Git è un sistema di controllo delle versioni distribuito e open source.
- HashiCorp Vault offre un'archiviazione sicura con accesso controllato per credenziali e altre informazioni sensibili.

- Helm è un gestore di pacchetti open source per Kubernetes che ti aiuta a installare e gestire le applicazioni sul tuo cluster Kubernetes.
- kubectl è un'interfaccia a riga di comando che ti aiuta a eseguire comandi sui cluster Kubernetes.
- Postman è una piattaforma API.

Archivio di codice

I file YAML di configurazione Kubernetes e gli script Python per questo modello sono disponibili nel repository. GitHub aws-patterns-edc Il pattern utilizza anche il repository Tractus-X EDC.

Best practice

Amazon EKS e isolamento delle infrastrutture dei partecipanti

Secondo questo schema, i namespace in Kubernetes separeranno l'infrastruttura del provider X dell'azienda dall'infrastruttura del consumatore dell'azienda Y. Per ulteriori informazioni, consulta le guide alle best practice di EKS.

In una situazione più realistica, ogni partecipante avrebbe un cluster Kubernetes separato in esecuzione all'interno del proprio. Account AWS L'infrastruttura condivisa (secondo questo modello DAPS) sarebbe accessibile dai partecipanti allo spazio dati pur essendo completamente separata dalle infrastrutture dei partecipanti.

Epiche

Configura l'ambiente ed esegui il provisioning di un cluster EKS e EC2 delle istanze

Operazione	Description	Competenze richieste
Clonare il repository.	Per clonare il repository sulla tua workstation, esegui il seguente comando: `git clone https://github.com/Think-iT-Labs/aws-patterns-edc` La workstation deve avere accesso al tuo. Account AWS	DevOps ingegnere

Operazione	Description	Competenze richieste

Esegui il provisioning del cluster Kubernetes e configura i namespace.	Per implementare un cluster EKS predefinito semplificato nel tuo account, esegui il seguente `eksctl` comando sulla workstation in cui hai clonato il repository:
	`eksctl create cluster`
	Il comando crea il VPC e le sottoreti private e pubbliche che si estendono su tre diverse zone di disponibilità. Dopo aver creato il livello di rete, il comando crea due `m5.large` EC2 istanze all'interno di un gruppo Auto Scaling.
	Per ulteriori informazioni ed esempi di output, consultate la guida eksctl.
	Dopo aver effettuato il provisioning del cluster privato, aggiungi il nuovo cluster EKS alla configurazione locale di Kubernetes eseguendo il comando seguente:
	`aws eks update-kubeconfig --name <EKS CLUSTER NAME> --region <AWS REGION>`
		DevOps ingegnere

Operazione	Description	Competenze richieste
	Questo modello utilizza il per eseguire tutti `eu-west-1` Regione AWS i comandi. Tuttavia, puoi eseguire gli stessi comandi nei tuoi comandi preferiti Regione AWS. Per confermare che i nodi EKS siano in esecuzion e e siano pronti, esegui il seguente comando: `kubectl get nodes`	

Configura i namespace.	Per creare namespace per il provider e il consumatore, esegui i seguenti comandi:
	
	`kubectl create ns provider`
	`kubectl create ns consumer`
	
	In questo schema, è important e utilizzare `provider` e `consumer` come namespace adattare le configurazioni dei passaggi successivi.
		DevOps ingegnere

Implementa il servizio di identità

Operazione	Description	Competenze richieste
Implementa DAPS utilizzando. AWS CloudFormation	Per facilitare la gestione delle operazioni DAPS, il server DAPS è installato sulle istanze. EC2 Per installare DAPS, usa il modello.AWS CloudFormation Avrai bisogno del certificato ACM e del nome DNS indicati nella sezione Prerequis iti. Il modello distribuisce e configura quanto segue: • Application Load Balancer • Gruppo Auto Scaling • EC2 istanze configurate con dati utente per installare tutti i pacchetti necessari • Ruoli IAM • DAPS È possibile distribuire il AWS CloudFormation modello accedendo Console di gestione AWS e utilizzando la AWS CloudFormation console. È inoltre possibile distribui re il modello utilizzando un AWS CLI comando come il seguente:	DevOps ingegnere

Operazione	Description	Competenze richieste
	`aws cloudformation create-stack --stack-name daps \ --template-body file://aws-patterns-edc/cloudformation.yml --parameters \ ParameterKey=CertificateARN,ParameterValue=<ACM Certificate ARN> \ ParameterKey=DNSName,ParameterValue=<DNS name> \ ParameterKey=InstanceType,ParameterValue=<EC2 instance type> \ ParameterKey=EnvironmentName,ParameterValue=<Environment Name> --capabilities CAPABILITY_NAMED_IAM`  Il nome dell'ambiente è a tua scelta. Ti consigliamo di utilizzare un termine significativo, ad esempio `DapsInfrastructure`, perché si rifletterà nei tag delle AWS risorse.  Per questo modello, `t3.small` è abbastanza grande da eseguire il flusso di lavoro DAPS, che ha tre contenitori Docker.	

Operazione	Description	Competenze richieste
	Il modello distribuisce le EC2 istanze in sottoreti private. Ciò significa che le istanze non sono accessibili direttamente tramite SSH (Secure Shell) da Internet. Alle istanze viene assegnato il ruolo e l' AWS Systems Manager agente IAM necessari per consentir e l'accesso alle istanze in esecuzione tramite Session Manager, una funzionalità di. AWS Systems Manager Si consiglia di utilizzar e Session Manager per l'accesso. In alternativa, puoi fornire un bastion host per consentire l'accesso SSH da Internet. Quando si utilizza l'approccio bastion host, l' EC2 istanza potrebbe impiegare qualche minuto in più per iniziare a funzionare. Dopo che il AWS CloudForm ation modello è stato distribui to correttamente, indica il nome DNS al nome DNS dell'Application Load Balancer. Per averne la conferma, esegui il comando seguente: `dig <DNS NAME>`	

Operazione	Description	Competenze richieste

	L’output visualizzato dovrebbe essere simile al seguente:
	
	`; <<>> DiG 9.16.1-Ubuntu <<>> edc-pattern.think-it.io`
	`;; global options: +cmd`
	`;; Got answer:`
	`;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 42344`
	`;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1`
	
	`;; OPT PSEUDOSECTION:`
	`; EDNS: version: 0, flags:; udp: 65494`
	`;; QUESTION SECTION:`
	`;edc-pattern.think-it.io. IN A`
	
	`;; ANSWER SECTION:`
	`edc-pattern.think-it.io. 276 IN CNAME daps-alb-iap9zmwy3kn8-1328773120.eu-west-1.elb.amazonaws.com.`
	`daps-alb-iap9zmwy3kn8-1328773120.eu-west-1.elb.amazonaws.com. 36 IN A 52.208.240.129`
	`daps-alb-iap9zmwy3kn8-1328773120.eu-west-1.elb.amazonaws.`	

Operazione	Description	Competenze richieste
	`com. 36 IN A 52.210.155.124`	

Operazione	Description	Competenze richieste

Registra i connettori dei partecipanti al servizio DAPS.	Dall'interno di una qualsiasi delle EC2 istanze previste per DAPS, registrate i partecipanti:
	1. Esegui lo script disponibile sull' EC2 istanza utilizzando l'utente root:
	`cd /srv/mvds/omejdn-daps`
	2. Registra il provider:
	`bash scripts/register_connector.sh <provider_name>`
	3. Registra il consumatore:
	`bash scripts/register_connector.sh <consumer_name>`
	La scelta dei nomi non influisce sui passaggi successivi. Ti consigliamo di utilizzare `provider` and `consumer` o `companyx` and`companyy`.
	I comandi di registrazione configureranno inoltre automaticamente il servizio DAPS con le informazioni necessarie recuperate dai certificati e dalle chiavi creati.
		DevOps ingegnere

Operazione	Description	Competenze richieste
	Mentre sei connesso a un server DAPS, raccogli le informazioni necessarie per le fasi successive dell'inst allazione: 1. Da `omejdn-daps/config/clients.yml` get the `client id` per il fornitore e il consumatore. I `client id` valori sono lunghe stringhe di cifre esadecimali. 2. Dalla `omejdn-daps/keys` directory, copia il contenuto dei file`consumer.cert` ,, `consumer.key` e. `provider.cert` `provider.key` Ti consigliamo di copiare e incollare il testo in file con nomi simili con il prefisso `daps-` sulla tua workstation. Dovresti avere il client IDs come fornitore e consumato re e dovresti avere quattro file nella cartella di lavoro sulla tua workstation: • Il nome del file di origine `consumer.cert` diventa il nome del file della	

Operazione	Description	Competenze richieste
	workstation. `daps-consumer.cert` • Il nome del file di origine `consumer.key` diventa il nome del file della workstation. `daps-consumer.key` • Il nome del file di origine `provider.cert` diventa il nome del file della workstation. `daps-provider.cert` • Il nome del file di origine `provider.key` diventa il nome del file della workstation. `daps-provider.key`	

Implementa i connettori dei partecipanti

Operazione	Description	Competenze richieste
Clona il repository Tractus-X EDC e usa la versione 0.4.1.	La costruzione del connettore Tractus-X EDC richiede l'implementazione e la disponibilità dei servizi PostgreSQL (database degli asset) e HashiCorp Vault (gestione dei segreti). Esistono molte versioni diverse dei grafici Tractus-X EDC Helm. Questo modello specifica la versione 0.4.1 perché utilizza il server DAPS.	DevOps ingegnere

Operazione	Description	Competenze richieste

	Le versioni più recenti utilizzano Managed Identity Wallet (MIW) con un'implementazione distribuita del servizio di identità.
	
	Sulla workstation in cui hai creato i due namespace Kubernetes, clona il repository tractusx-edc ed esamina il ramo. `release/0.4.1`
	
	`git clone https://github.com/eclipse-tractusx/tractusx-edc`
	
	`cd tractusx-edc`
	
	`git checkout release/0.4.1`	

Operazione	Description	Competenze richieste

Configura la tabella Tractus-X EDC Helm.	Modifica la configurazione del modello di grafico Tractus-X Helm per consentire a entrambi i connettori di interagire tra loro.
	
	A tale scopo, è necessario aggiungere lo spazio dei nomi al nome DNS del servizio in modo che possa essere risolto da altri servizi del cluster. Queste modifiche devono essere apportate al file. `charts/tractusx-connector/templates/_helpers.tpl` Questo modello fornisce una versione finale modificata di questo file da utilizzare. Copialo e inseriscilo nella `daps` sezione del file`charts/tractusx-connector/templates/_helpers.tpl` .
	
	Assicurati di commentare tutte le dipendenze DAPS in: `charts/tractusx-connector/Chart.yaml`
	
	`dependencies:`
	`  # IDS Dynamic Attribute Provisioning Service (IAM)`
	`  #  - name: daps`
	`  # version: 0.0.1`
		DevOps ingegnere

Operazione	Description	Competenze richieste

	`  # repository: "file://./subcharts/omejdn"`
	`  # alias: daps`
	`  # condition: install.daps`	

Operazione	Description	Competenze richieste

Configura i connettori per utilizzare PostgreSQL su Amazon RDS.	(Facoltativo) L'istanza di Amazon Relational Database Service (Amazon RDS) non è richiesta in questo modello. Tuttavia, consigliamo vivamente di utilizzare Amazon RDS o Amazon Aurora perché offrono funzionalità come disponibilità elevata e backup e ripristino.
	
	Per sostituire PostgreSQL su Kubernetes con Amazon RDS, procedi come segue:
	
	1. Effettua il provisioning dell'istanza Amazon RDS for PostgreSQL.
	2. Nel, commenta `Chart.yaml` la sezione. `PostgreSQL`
	3. In `provider_values.yml` `econsumer_values.yml` , configura la `postgresql` sezione come segue:
	
	`postgresql:`
	`  auth:`
	`  database: edc`
	`  password: <RDS PASSWORD>`
	`  username: <RDS Username>`
		DevOps ingegnere

Operazione	Description	Competenze richieste

	<pre> jdbcUrl: jdbc:post
	gresql://<RDS DNS
	 NAME>:5432/edc
	  username: <RDS
	 Username>
	  password: <RDS
	 PASSWORD>
	  primary:
	  persistence:
	  enabled: false
	  readReplicas:
	  persistence:
	  enabled: false</pre>	

Operazione	Description	Competenze richieste
Configura e distribuisci il connettore del provider e i relativi servizi.	Per configurare il connettore del provider e i relativi servizi, procedi come segue: 1. Per scaricare il `provider_edc.yaml` file dalla `edc_helm_configs` directory alla cartella Helm chart corrente, esegui il comando seguente: `wget -q https://raw.githubusercontent.com/Think-iT-Labs/aws-patterns-edc/main/edc_helm_configs/provider_edc.yaml> -P charts/tractusx-connector/` 2. Sostituisci le seguenti variabili (anch'esse contrassegnate nel file) con i relativi valori: • `CLIENT_ID` – L'ID generato dal DAPS. `CLIENT_ID` Dovrebbe essere presente `/srv/mvds/omejdn-daps/config/clients.yml/config/clients.yml` sul server DAPS. Dovrebbe essere una stringa di caratteri esadecimali.	DevOps ingegnere

Operazione	Description	Competenze richieste

	• DAPS_URL– L'URL del server DAPS. Dovrebbe `https://{DNS name}` usare il nome DNS che hai impostato quando hai eseguito il AWS CloudFormation modello.
	• `VAULT_TOKEN` – Il token da utilizzare per l'autorizzazione Vault. Scegli un valore qualsiasi.
	• `vault.fullnameOverride` – `vault-provider` .
	• `vault.hashicorp.url` – `http://vault-provider:8200/` .
	
	I valori precedenti presuppongono che il nome della distribuzione e il nome dello spazio dei nomi siano provider.
	3. Per eseguire il grafico Helm dalla tua workstation, usa i seguenti comandi:
	
	`cd charts/tractusx-connector`
	
	`helm dependency build`
	
	`helm upgrade --install provider ./ -f`	

Operazione	Description	Competenze richieste
	`provider_edc.yaml -n provider`	

Operazione	Description	Competenze richieste

Aggiungi il certificato e le chiavi al vault del provider.	Per evitare confusione, produci i seguenti certificati al di fuori della `tractusx-edc/charts` directory.
	
	Ad esempio, esegui il comando seguente per passare alla tua home directory:
	
	`cd ~`
	
	Ora è necessario aggiungere i segreti necessari al provider nel vault.
	
	I nomi dei segreti all'interno del vault sono i valori delle chiavi nella `secretNames:` sezione del `provider_edc.yml` file. Per impostazi one predefinita, sono configurati come segue:
	
	`secretNames: transferProxyTokenSignerPrivateKey: transfer-proxy-token-signer-private-key transferProxyTokenSignerPublicKey: transfer-proxy-token-signer-public-key transferProxyTokenEncryptio`
		DevOps ingegnere

Operazione	Description	Competenze richieste

	`nAesKey: transfer-proxy-token-encryption-aes-key`
	`dapsPrivateKey: daps-private-key`
	`dapsPublicKey: daps-public-key`
	
	Inizialmente vengono generati una chiave AES (Advanced Encryption Standard), una chiave privata, una chiave pubblica e un certificato autofirmato. Questi vengono successivamente aggiunti come segreti al vault.
	
	Inoltre, questa directory dovrebbe contenere i `daps-provider.key` file `daps-provider.cert` e che avete copiato dal server DAPS.
	
	1. Esegui i comandi seguenti:
	
	`# generate a private key`
	`openssl ecparam -name prime256v1 -genkey -noout -out provider-private-key.pem`
	`# generate corresponding public key`
	`openssl ec -in provider-private-key.pem -pubout -out`	

Operazione	Description	Competenze richieste
	``` provider-public-ke y.pem # create a self-sign ed certificate openssl req -new - x509 -key provider- private-key.pem -out provider-cert.pem - days 360 # generate aes key openssl rand -base64 32 > provider- aes.key ```	
	2. Prima di aggiungere i segreti al vault, convertit eli da righe multiple a righe singole sostituendo le interruzioni di riga con: \n ``` cat provider-private- key.pem | sed 's/$/\ \\\n/'\|tr -d '\\n' > provider-private-k ey.pem.line cat provider-public- key.pem \| sed 's/$/\ \\\n/'|tr -d '\\n' > provider-public-ke y.pem.line cat provider-cert.pem | sed 's/$/\\\\ n/'\|tr -d '\\n' > provider-cert.pem. line cat provider-aes.key \| sed 's/$/\\\\ n/'|tr -d '\\n' > provider-aes.key.l ine ```	

Operazione	Description	Competenze richieste
	``` ## The following block is for daps certifica te and key openssl x509 -in daps-provider.cert - outform PEM | sed 's/ $/\\\\n/'\|tr -d '\\n' > daps-provider.cert .line cat daps-provider.key \| sed 's/$/\\\\n/'| tr -d '\\n' > daps- provider.key.line ``` 3. Per formattare i segreti che verranno aggiunti a Vault, esegui i seguenti comandi: ``` JSONFORMAT='{"cont ent": "%s"}' #create a single line in JSON format printf "${JSONFO RMAT}\\n" "`cat  provider-private- key.pem.line`" > provider-private-k ey.json printf "${JSONFO RMAT}\\n" "`cat  provider-public- key.pem.line`" > provider-public-ke y.json printf "${JSONFO RMAT}\\n" "`cat  provider-cert.pem. line`" > provider- cert.json printf "${JSONFO RMAT}\\n" "`cat ```	

Operazione	Description	Competenze richieste
	provider-aes.key.line`" > provider-aes.json  printf "${JSONFORMAT}\\n" "`cat daps-provider.key.line`" > daps-provider.key.json printf "${JSONFORMAT}\\n" "`cat daps-provider.cert.line`" > daps-provider.cert.json  I segreti sono ora in formato JSON e sono pronti per essere aggiunti al vault. 4. Per ottenere il nome del pod per il vault, esegui il seguente comando:  kubectl get pods -n provider|egrep "vault|NAME"  Il nome del pod sarà simile a. `"vault-provider-0"` Questo nome viene utilizzato quando si crea un port forward verso il vault. Il port forward consente di accedere al vault per aggiungere il segreto. Dovresti eseguirlo da una	

Operazione	Description	Competenze richieste
	workstation con credenziali AWS configurate. 5. Per accedere al vault, usa `kubectl` per configurare un port forward: `kubectl port-forward <VAULT_POD_NAME> 8200:8200 -n provider` Ora dovresti essere in grado di accedere al vault tramite il tuo browser o la CLI. Browser 1. Utilizzando il browser, accedi a http://127.0.0.1:8200, che utilizzerà il port forward che hai configurato. 2. Accedi utilizzando il token che hai configurato in precedenza`provider_edc.yml` . Nel motore dei segreti, crea tre segreti. Ogni segreto avrà un `Path for this secret` valore, che è il nome segreto mostrato nell'elenco seguente. All'interno della `secret data` sezione, il nome della chiave sarà `content` e il valore sarà la singola riga di testo del	

Operazione	Description	Competenze richieste
	rispettivo file denominat o.line. 3. I nomi segreti provengon o dalla secretNames sezione del provider_ edc.yml file. 4. Crea i seguenti segreti: • Segreto transfer- proxy-token- signer-private- key con nome di file provider-private-k ey.pem.line • Segreto transfer- proxy-token- signer-public-k ey con nome di file provider-cert.pem. line • Segreto transfer- proxy-token- encryption-aes- key con nome di file provider-aes.key.l ine • Segreto daps-priv ate-key con nome di file daps-prov ider.key.line • Segreto daps-publ ic-key con nome di file daps-provider.cert .line	

Operazione	Description	Competenze richieste

	CLI di Vault
	
	La CLI utilizzerà anche il port forward che hai configurato.
	
	1. Sulla tua workstation, installa Vault CLI seguendo le istruzioni nella documentazione di Vault. HashiCorp
	2. Per accedere al vault utilizzando il token che hai configuratoprovider_edc.yml , esegui il seguente comando:
	
	`vault login -address=http://127.0.0.1:8200`
	
	Con il token corretto, dovresti vedere il messaggio "Success! You are now authenticated."
	3. Per creare i segreti utilizzando i file in formato JSON creati in precedenza, esegui il codice seguente:
	
	`vault kv put -address=http://127.0.0.1:8200 secret/transfer-proxy-token-signer-private-key @provider-private-key.json`	

Operazione	Description	Competenze richieste

	`vault kv put -address=http://127.0.0.1:8200 secret/transfer-proxy-token-signer-public-key @provider-cert.json`
	`vault kv put -address=http://127.0.0.1:8200 secret/transfer-proxy-token-encryption-aes-key @provider-aes.json`
	
	`vault kv put -address=http://127.0.0.1:8200 secret/daps-private-key @daps-provider.key.json`
	`vault kv put -address=http://127.0.0.1:8200 secret/daps-public-key @daps-provider.cert.json`	

Operazione	Description	Competenze richieste

Configura e implementa il Consumer Connector e i relativi servizi.	I passaggi per la configura zione e la distribuzione del consumer sono simili a quelli che hai completato per il provider:
	
	1. Per copiarlo `consumer_edc.yaml` dal aws-patterns-edcrepository nella cartella `tractusx-edc/charts/tractusx-connecto r`, esegui i seguenti comandi:
	
	`cd tractusx-edc`
	
	`wget -q https://raw.githubusercontent.com/Think-iT-Labs/aws-patterns-edc/main/edc_helm_configs/consumer_edc.yaml -P charts/tractusx-connector/`
	
	2. Aggiorna le seguenti variabili con i loro valori effettivi:
	
	• `CONSUMER_CLIENT_ID` – L'ID generato da DAPS. `CONSUMER_CLIENT_ID` Dovrebbe essere presente `config/clients.yml` sul server DAPS.	

Operazione	Description	Competenze richieste

	• `DAPS_URL`– Lo stesso URL DAPS utilizzato per il provider.
	• `VAULT_TOKEN` – Il token da utilizzare per l'autorizzazione Vault. Scegli un valore qualsiasi.
	• `vault.fullnameOverride` – `vault-consumer`
	• `vault.hashicorp.url` – `http://vault-provider:8200/`
	
	I valori precedenti presuppongono che il nome della distribuzione e il nome dello spazio dei nomi siano. `consumer`
	3. Per eseguire il grafico Helm, utilizzare i seguenti comandi:
	
	`cd charts/tractusx-connector`
	
	`helm upgrade --install consumer ./ -f consumer_edc.yaml -n consumer`	

Operazione	Description	Competenze richieste
Aggiungi il certificato e le chiavi al Consumer Vault.	Dal punto di vista della sicurezza, consigliamo di rigenerare i certificati e le chiavi per ogni partecipante allo spazio dati. Questo modello rigenera certificati e chiavi per il consumatore. I passaggi sono molto simili a quelli del provider. È possibile verificare i nomi segreti contenuti nel `consumer_edc.yml` file. I nomi dei segreti all'interno del vault sono i valori delle chiavi nella `secretNames:` sezione di. `consumer_edc.yml file` Per impostazione predefinita, sono configurati come segue: `secretNames: transferProxyTokenSignerPrivateKey: transfer-proxy-token-signer-private-key transferProxyTokenSignerPublicKey: transfer-proxy-token-signer-public-key transferProxyTokenEncryptionAesKey: transfer-proxy-token-encryption-aes-key`	DevOps ingegnere

Operazione	Description	Competenze richieste

	<pre> dapsPriva
	teKey: daps-private-
	key
	                dapsPubli
	cKey: daps-public-key</pre>
	I `daps-consumer.key` file `daps-consumer.cert` and copiati dal server DAPS dovrebbero già esistere in questa directory.
	1. Esegui i comandi seguenti:
	<pre># generate a private
	 key
	openssl ecparam -name
	 prime256v1 -genkey
	 -noout -out consumer-
	private-key.pem
	# generate correspon
	ding public key
	openssl ec -in
	 consumer-private-k
	ey.pem -pubout -out
	 consumer-public-ke
	y.pem
	# create a self-sign
	ed certificate
	openssl req -new -
	x509 -key consumer-
	private-key.pem -out
	 consumer-cert.pem -
	days 360
	# generate aes key
	openssl rand -base64
	 32 > consumer-
	aes.key</pre>	

Operazione	Description	Competenze richieste

	2. Modifica manualmente i file per sostituire le interruzioni di \n riga o usa tre comandi simili ai seguenti:
	
	`cat consumer-private-key.pem | sed 's/$/\\\n/'|tr -d '\\n' > consumer-private-key.pem.line`
	`cat consumer-public-key.pem | sed 's/$/\\\n/'|tr -d '\\n' > consumer-public-key.pem.line`
	`cat consumer-cert.pem | sed 's/$/\\\\n/'|tr -d '\\n' > consumer-cert.pem.line`
	`cat consumer-aes.key | sed 's/$/\\\\n/'|tr -d '\\n' > consumer-aes.key.line`
	`cat daps-consumer.cert | sed 's/$/\\\\n/'|tr -d '\\n' > daps-consumer.cert.line`
	`cat daps-consumer.key | sed 's/$/\\\\n/'| tr -d '\\n' > daps-consumer.key.line`
	
	3. Per formattare i segreti che verranno aggiunti a Vault, esegui i seguenti comandi:	

Operazione	Description	Competenze richieste

	`JSONFORMAT='{"content": "%s"}'`
	
	`#create a single line in JSON format`
	
	`printf "${JSONFORMAT}\\n" "`cat consumer-private-key.pem.line`" > consumer-private-key.json`
	`printf "${JSONFORMAT}\\n" "`cat consumer-public-key.pem.line`" > consumer-public-key.json`
	`printf "${JSONFORMAT}\\n" "`cat consumer-cert.pem.line`" > consumer-cert.json`
	`printf "${JSONFORMAT}\\n" "`cat consumer-aes.key.line`" > consumer-aes.json`
	
	`printf "${JSONFORMAT}\\n" "`cat daps-consumer.key.line`" > daps-consumer.key.json`
	`printf "${JSONFORMAT}\\n" "`cat daps-consumer.cert.line`" > daps-consumer.cert.json`	

Operazione	Description	Competenze richieste

	I segreti sono ora in formato JSON e sono pronti per essere aggiunti al vault.
	4. Per ottenere il nome del pod per il consumer vault, esegui il seguente comando:
	`kubectl get pods -n consumer | egrep "vault|NAME"`
	Il nome del pod sarà simile a. `"vault-consumer-0"` Questo nome viene utilizzato quando si crea un port forward verso il vault. Il port forward consente di accedere al vault per aggiungere il segreto. È necessario eseguirlo da una workstation con AWS credenziali configurate.
	5. Per accedere al vault, usa `kubectl` per configurare un port forward:
	`kubectl port-forward <VAULT_POD_NAME> 8201:8200 -n consumer`
	Questa volta la porta locale è 8201, quindi è possibile	

Operazione	Description	Competenze richieste
	utilizzare i port forward sia per il produttore che per il consumatore. Browser Puoi usare il tuo browser per connetterti a http://localhost:8201/ per accedere al Consumer Vault e creare i segreti con nomi e contenuti come indicato. I segreti e i file che contengono il contenuto sono i seguenti: • Segreto `transfer-proxy-token-signer-private-key` con nome di file `consumer-private-key.pem.line` • Segreto `transfer-proxy-token-signer-public-key` con nome di file `consumer-cert.pem.line` • Segreto `transfer-proxy-token-encryption-aes-key` con nome di file `consumer-aes.key.line` CLI di Vault	

Operazione	Description	Competenze richieste
	Utilizzando la CLI di Vault, è possibile eseguire i seguenti comandi per accedere al vault e creare i segreti: 1. Accedi al vault utilizzando il token che hai configurato all'interno: `consumer_edc.yml` ``` vault login -address= http://127.0.0.1:8 201 ``` Con il token corretto, dovresti vedere il messaggio `"Success! You are now authenticated."` 2. Per creare i segreti utilizzando i file in formato JSON creati in precedenza, esegui il codice seguente: ``` vault kv put -address= http://127.0.0.1:8 201 secret/transfer- proxy-token-signer-p rivate-key @consumer -private-key.json vault kv put - address=http://12 7.0.0.1:8201 secret/ transfer-proxy-token -signer-public-key @consumer-cert.json vault kv put -address= http://127.0.0.1:8	

Operazione	Description	Competenze richieste
	201 secret/transfer-proxy-token-encryption-aes-key @consumer-aes.json vault kv put -address=http://127.0.0.1:8201 secret/daps-private-key @daps-consumer.key.json vault kv put -address=http://127.0.0.1:8201 secret/daps-public-key @daps-consumer.cert.json	

Configura un client HTTP per interagire con l'API di gestione dei connettori

Operazione	Description	Competenze richieste

Configura il port forwarding.	1. Per controllare lo stato dei pod, esegui i seguenti comandi:
	`kubectl get pods -n provider`
	`kubectl get pods -n consumer`
	2. Per assicurarti che le implementazioni di Kubernetes abbiano avuto successo, guarda i log dei pod Kubernetes del provider e del consumatore eseguendo i seguenti comandi:
		DevOps ingegnere

Operazione	Description	Competenze richieste

	`kubectl logs -n provider <producer control plane pod name>`
	`kubectl logs -n consumer <consumer control plane pod name>`
	
	Il cluster è privato e non è accessibile pubblicamente. Per interagire con i connettori, utilizza la funzionalità di port forwarding di Kubernetes per inoltrare il traffico generato dalla macchina al piano di controllo del connettore.
	
	1. Sul primo terminale, inoltra le richieste del consumatore all'API di gestione tramite la porta 8300:
	
	`kubectl port-forward deployment/consumer-tractusx-connector-controlplane 8300:8081 -n consumer`
	
	2. Sul secondo terminale, inoltra le richieste del provider all'API di gestione tramite la porta 8400:
	
	`kubectl port-forward deployment/provide`	

Operazione	Description	Competenze richieste
	`r-tractusx-connect or-controlplane 8400:8081 -n provider`	

Operazione	Description	Competenze richieste
Crea bucket S3 per il provider e il consumatore.	Il connettore EDC attualmente non utilizza credenziali AWS temporanee, come quelle fornite assumendo un ruolo. L'EDC supporta solo l'uso di un ID di chiave di accesso IAM e una combinazione di chiavi di accesso segrete. Per le fasi successive sono necessari due bucket S3. Un bucket S3 viene utilizzato per archiviare i dati resi disponibili dal provider. L'altro bucket S3 è per i dati ricevuti dal consumatore. L'utente IAM deve avere il permesso di leggere e scrivere oggetti solo nei due bucket denominati. È necessario creare e proteggere un ID della chiave di accesso e una coppia di chiavi di accesso segrete. Dopo la disattivazione di questo MVDS, l'utente IAM deve essere eliminato. Il codice seguente è un esempio di policy IAM per l'utente: {	DevOps ingegnere

Operazione	Description	Competenze richieste
	"Version": "2012-10-17", "Statement": [{ "Sid": "Stmt1708699805237", "Action": ["s3:GetObject", "s3:GetObjectVersion", "s3:ListAllMyBuckets", "s3:ListBucket", "s3:ListBucketMultipartUploads", "s3:ListBucketVersions", "s3:PutObject"], "Effect": "Allow", "Resource": ["arn:aws:s3:::<S3 Provider Bucket>", "arn:aws:s3:::<S3 Consumer Bucket>", "arn:aws:s3:::<S3 Provider Bucket>/*", "arn:aws:s3:::<S3 Consumer Bucket>/*"] }] }	

Operazione	Description	Competenze richieste
Configura Postman per interagire con il connettore.	Ora puoi interagire con i connettori tramite la tua EC2 istanza. Usa Postman come client HTTP e fornisci Postman Collections sia per il provider che per i connettori consumer. Importa le raccolte dal `aws-pattern-edc` repository nella tua istanza di Postman. Questo modello utilizza le variabili di raccolta Postman per fornire input alle richieste.	Sviluppatore di app, ingegnere dei dati

Fornisci i dati sull'impronta di carbonio dell'azienda X tramite il connettore

Operazione	Description	Competenze richieste
Prepara i dati sull'intensità delle emissioni di carbonio da condividere.	Per prima cosa devi decidere quale asset di dati condividere. I dati dell'azienda X rappresentano l'impronta delle emissioni di carbonio della sua flotta di veicoli. Il peso è il peso lordo del veicolo (GVW) in tonnellate e le emissioni sono espresse in grammi CO2 per tonnellata-chilometro (g CO2 e/t-km) secondo la misurazione (WTW): Wheel-to-Well	Ingegnere dei dati, sviluppatore di app

Operazione	Description	Competenze richieste
	• Tipo di veicolo: furgone; peso: < 3,5; emissioni: 800 • Tipo di veicolo: autocarro urbano; peso: 3,5/7,5; emissioni: 315 • Tipo di veicolo: veicolo per il trasporto di merci di medie dimensioni (MGV); peso: 7,5/20; emissioni: 195 • Tipo di veicolo: veicolo commerciale pesante (HGV); peso: > 20; emissioni: 115 I dati di esempio si trovano nel `carbon_emissions_data.json` file del repository. `aws-patterns-edc` Company X utilizza Amazon S3 per archiviare oggetti. Crea il bucket S3 e archivia lì l'oggetto dati di esempio. I seguenti comandi creano un bucket S3 con impostazioni di sicurezza predefinite. Consigliamo vivamente di consultare le best practice di sicurezza per Amazon S3. `aws s3api create-bucket <BUCKET_NAME> --region <AWS_REGION>`	

Operazione	Description	Competenze richieste
	``` # You need to add '--create-bucket-c onfiguration # LocationConstraint =<AWS_REGION>' if you want to create # the bucket outside of us- east-1 region aws s3api put-object --bucket <BUCKET_NAME> \ --key <S3 OBJECT NAME> \ --body <PATH OF THE FILE TO UPLOAD> ``` Il nome del bucket S3 deve essere univoco a livello globale. Per ulteriori informazioni sulle regole di denominazione, consulta la documentazione AWS.	

Operazione	Description	Competenze richieste
Registra la risorsa di dati nel connettore del provider utilizzando Postman.	Un asset di dati del connettore EDC contiene il nome dei dati e la loro posizione. In questo caso, l'asset di dati del connettore EDC punterà all'oggetto creato nel bucket S3: • Connettore: Provider • Richiesta: Crea risorsa • Variabili di raccolta: aggiornamento `ASSET_NAME`. Scegliete un nome significativo che rappresenti la risorsa. • Corpo della richiesta: aggiorna il corpo della richiesta con il bucket S3 che hai creato per il provider. `"dataAddress": { "edc:type": "AmazonS3", "name": "Vehicle Carbon Footprint", "bucketName": "<REPLACE WITH THE SOURCE BUCKET NAME>", "keyName": "<REPLACE WITH YOUR OBJECT NAME>", "region": "<REPLACE WITH THE BUCKET REGION>",`	Sviluppatore di app, ingegnere dei dati

Operazione	Description	Competenze richieste
	<pre>"accessKeyId": "<REPLACE WITH YOUR ACCESS KEY ID>", "secretAccessKey": "<REPLACE WITH SECRET ACCESS KEY>" }</pre> • Risposta: Una richiesta riuscita restituisce l'ora di creazione e l'ID dell'asset appena creato. <pre>{ "@id": "c89aa31c-ec4c-44ed-9e8c-1647f19d7583" }</pre> • Variabile di raccolta ASSET_ID: aggiorna la variabile di raccolta Postman `ASSET_ID` con l'ID generato automaticamente dal connettore EDC dopo la creazione.	

Operazione	Description	Competenze richieste

Definisci la politica di utilizzo della risorsa.	Un asset di dati EDC deve essere associato a politiche di utilizzo chiare. Innanzitutto, crea la definizione della politica nel connettore del provider.
	
	La politica dell'azienda X consiste nel consentire ai partecipanti allo spazio dati di utilizzare i dati sull'impronta delle emissioni di carbonio.
	
	• Organo della richiesta:
	  • Connettore: Provider
	  • Richiesta: creazione di una politica
	  • Variabili di raccolta: aggiorna la `Policy Name` variabile con il nome della politica.
	• Risposta: Una richiesta riuscita restituisce l'ora di creazione e l'ID della politica appena creata. Aggiorna la variabile di raccolta `POLICY_ID` con l'ID della politica generata dal connettore EDC dopo la creazione.
		Sviluppatore di app, ingegnere dei dati

Operazione	Description	Competenze richieste
Definisci un'offerta contrattuale EDC per l'asset e la sua politica di utilizzo.	Per consentire agli altri partecipanti di richiedere l'accesso ai tuoi dati, offrilo in un contratto che specifichi le condizioni di utilizzo e le autorizzazioni: • Connettore: Provider • Richiesta: creazione della definizione del contratto • Variabili di raccolta: aggiorna la `Contract Name` variabile con un nome per l'offerta o la definizione del contratto.	Sviluppatore di app, ingegnere dei dati

Scopri gli asset e raggiungi un accordo sui contratti definiti

Operazione	Description	Competenze richieste
Richiedi il catalogo dati condiviso dalla società X.	In qualità di consumatore di dati nello spazio dati, l'azienda Y deve innanzitutto scoprire i dati condivisi dagli altri partecipanti. In questa configurazione di base, puoi farlo chiedendo al consumer connector di richiedere il catalogo delle risorse disponibili direttamente dal connettore del provider. • Connettore: Consumer	Sviluppatore di app, ingegnere dei dati

Operazione	Description	Competenze richieste

	• Richiesta: Richiedi il catalogo
	• Risposta: tutte le risorse di dati disponibili del provider insieme alle politiche di utilizzo allegate. In qualità di consumatore di dati, cerca il contratto che ti interessa e aggiorna di conseguenza le seguenti variabili di raccolta.
	  • `CONTRACT_OFFER_ID` – L'ID dell'offerta contrattuale che il consumatore desidera negoziare
	  • `ASSET_ID`– L'ID dell'asset che il consumatore desidera negoziare
	  • `PROVIDER_CLIENT_ID` – L'ID del connettore del provider con cui negoziare	

Operazione	Description	Competenze richieste
Avvia una negoziazione contrattuale per i dati sull'intensità delle emissioni di carbonio della società X.	Ora che avete identificato l'asset che desiderate consumare, avviate un processo di negoziazione del contratto tra il consumatore e il fornitore Connectors. • Connettore: Consumatore • Richiesta: negoziazione del contratto • Variabili di raccolta: aggiorna la `CONSUMER_CLIENT_ID` variabile con l'ID del connettore consumer con cui negoziare. Il processo potrebbe richiedere del tempo prima di raggiungere lo stato VERIFIED. È possibile verificare lo stato della negoziazione del contratto e l'ID dell'accordo corrispondente utilizzando la `Get Negotiation` richiesta.	Sviluppatore di app, ingegnere dei dati

Consuma i dati utilizzando il contratto

Operazione	Description	Competenze richieste
Consuma dati dagli endpoint HTTP.	(Opzione 1) Per utilizzare il piano dati HTTP per	Sviluppatore di app, ingegnere dei dati

Operazione	Description	Competenze richieste
	consumare i dati nello spazio dati, puoi utilizzare webhook.site per emulare un server HTTP e avviare il processo di trasferimento nel consumer connector: • Connettore: Consumer • Richiesta: negoziazione del contratto • Variabili di raccolta: aggiorna la `Contract Agreement ID` variabile con l'ID del contratto generato dal connettore EDC. • Corpo della richiesta: aggiorna il corpo della richiesta specificandoHTTP, `dataDestination` oltre all'URL del webhook: `{ "dataDestination": { "type": "HttpProxy" }, "privateProperties": { "receiverHttpEndpoint": "<WEBHOOK URL>" } }`	

Operazione	Description	Competenze richieste
	Il connettore invierà le informazioni necessarie per scaricare il file direttamente all'URL del webhook. Il payload ricevuto è simile al seguente: `{ "id": "dcc90391-3819-4b54-b401-1a005a029b78", "endpoint": "http://consumer-tractusx-connector-dataplane.consumer:8081/api/public", "authKey": "Authorization", "authCode": "<AUTH CODE YOU RECEIVE IN THE ENDPOINT>", "properties": { "https://w3id.org/edc/v0.0.1/ns/cid": "vehicle-carbon-footprint-contract:4563abf7-5dc7-4c28-bc3d-97f45e32edac:b073669b-db20-4c83-82df-46b583c4c062" } }` Utilizza le credenziali ricevute per ottenere la risorsa S3 condivisa dal provider.	

Operazione	Description	Competenze richieste
	In quest'ultimo passaggio, è necessario inviare la richiesta al piano dati del consumatore (inoltrare correttamente le porte), come indicato nel payload (). `endpoint`	

Operazione	Description	Competenze richieste
Consuma direttamente i dati dai bucket S3.	(Opzione 2) Utilizza l'integrazione di Amazon S3 con il connettore EDC e punta direttamente al bucket S3 nell'infrastruttura consumer come destinazione: • Request Body: aggiorna il corpo della richiesta per specificare il bucket S3 come DataDestination. Dovrebbe essere il bucket S3 creato in precedenza per archiviare i dati ricevuti dal consumatore. `{ "dataDestination": { "type": "AmazonS3", "bucketName": "{{ REPLACE WITH THE DESTINATION BUCKET NAME }}", "keyName": "{{ REPLACE WITH YOUR OBJECT NAME }}", "region": "{{ REPLACE WITH THE BUCKET REGION }}", "accessKeyId": "{{ REPLACE WITH YOUR ACCESS KEY ID }}", "secretAccessKey": "{{ REPLACE WITH SECRET ACCESS KEY }}"`	Ingegnere dei dati, sviluppatore di app

Operazione	Description	Competenze richieste

	<pre>}
	}
	}</pre>	

Risoluzione dei problemi

Problema	Soluzione
Il connettore potrebbe sollevare un problema relativo al formato PEM del certificato.	Concatena il contenuto di ogni file in una singola riga aggiungendo. \n

Risorse correlate

- DSSC
- Creazione di spazi di dati per casi d'uso di sostenibilità (strategia AWS Prescriptive Guidance di Think-IT)
- AWS per spazi dati
- Documentazione Tractus-X
- DAPS
- Abilitare la condivisione dei dati tramite spazi dati e AWS (post sul blog)

Informazioni aggiuntive

Specifiche dello spazio dati

Partecipanti

Partecipante	Descrizione dell'azienda	Focus dell'azienda
Azienda X	Gestisce una flotta di veicoli in Europa e Sud America per il trasporto di merci varie.	Mira a prendere decisioni basate sui dati per ridurre

		l'intensità delle emissioni di carbonio.
Azienda Y	Un'autorità di regolamentazione ambientale	Applica le normative e le politiche ambientali progettate per monitorare e mitigare l'impatto ambientale di aziende e industrie, compresa l'intensità delle emissioni di carbonio.

Caso aziendale

La società X utilizza la tecnologia dello spazio dati per condividere i dati sull'impronta di carbonio con un revisore della conformità, la società Y, per valutare e affrontare l'impatto ambientale delle operazioni logistiche dell'azienda X.

Autorità per lo spazio dei dati

L'autorità dello spazio dati è un consorzio di organizzazioni che governano lo spazio dati. In questo modello, sia la società X che la società Y costituiscono l'organo di governo e rappresentano un'autorità federata per lo spazio dei dati.

Componenti dello spazio dati

Componente	Implementazione scelta	Informazioni aggiuntive
Protocollo di scambio di set di dati	Protocollo Dataspace versione 0.8	• JSON-LD • Vocabolario del catalogo dati (DCAT)
Connettore per spazio dati	Connettore Tractus-X EDC versione 0.4.1	• Estensioni EDC
Politiche di scambio di dati	Politica USE predefinita	• Open Digital Rights Language (ODRL)

Servizi di spazio dati

Servizio	Implementazione	Informazioni aggiuntive
Servizio di identità	Sistema di approvvigionamento dinamico degli attributi (DAPS)	«Un Dynamic Attribute Provisioning System (DAPS) ha l'intento di accertare determinati attributi delle organizzazioni e dei connettori. Pertanto, le terze parti non devono fidarsi di questi ultimi a condizione che si fidino delle affermazioni DAPS». — DAPS Per concentrarsi sulla logica del connettore, lo spazio dati viene distribuito su una EC2 macchina Amazon utilizzando Docker Compose.
Servizio di individuazione	Catalogo federato Gaia-X	«Il catalogo federato costituisce un archivio indicizzato di Gaia-X Self-Descriptions per consentire la scoperta e la selezione dei fornitori e delle loro offerte di servizi. Le autodescrizioni sono le informazioni fornite dai partecipanti su se stessi e sui loro servizi sotto forma di proprietà e reclami». — Kickstarter dell'ecosistema Gaia-X

Dati da scambiare

Risorse di dati	Description	Formato

Dati sulle emissioni di carbonio	Valori di intensità per diversi tipi di veicoli nella regione specificata (Europa e Sud America) dell'intera flotta di veicoli	File JSON

Modello di dati

```
{
  "region": "string",
  "vehicles": [
// Each vehicle type has its Gross Vehicle Weight (GVW) category and its emission
 intensity in grams of CO2 per Tonne-Kilometer (g CO2 e/t-km) according to the "Well-
to-Wheel" (WTW) measurement.
    {
      "type": "string",
      "gross_vehicle_weight": "string",
      "emission_intensity": {
        "CO2": "number",
        "unit": "string"
      }
    }
  ]
}
```

Connettore Tractus-X EDC

Per la documentazione di ogni parametro Tractus-X EDC, consulta il file dei valori originali.

La tabella seguente elenca tutti i servizi, insieme alle porte e agli endpoint esposti corrispondenti come riferimento.

Nome del servizio	Porta e percorso
Piano di controllo	• gestione: – Porta: 8081 Percorso: `/management` • controllo – Porta: 8083 Percorso: `/control`

	• Porta del protocollo: 8084 Percorso: `/api/v1/dsp` • metriche – Porta: 9090 Percorso: `/metrics` • osservabilità – Porta: 8085 Percorso: `/observability`
Piano dati	impostazione predefinita – Porta: 8080 Percorso: `/api` public – Porta: 8081 Percorso: `/api/dataplane/control` proxy – Porta: 8186 Percorso: `/proxy` metriche – Porta: 9090 Percorso: `/metrics` osservabilità – Porta: 8085 Percorso: `/observability`
Vault	Porta: 8200
PostgreSQL	Porta: 5432

Utilizzo di Manager Gestione dei segreti AWS

È possibile utilizzare Secrets Manager anziché HashiCorp Vault come gestore dei segreti. Per farlo, devi usare o creare l'estensione Gestione dei segreti AWS EDC.

Sarai responsabile della creazione e della manutenzione della tua immagine, perché Tractus-X non fornisce supporto per Secrets Manager.

Per fare ciò, è necessario modificare i file Gradle di compilazione sia del piano di controllo che del piano dati del connettore introducendo l'estensione Gestione dei segreti AWS EDC (vedi questo artefatto di Maven per un esempio), quindi creare, gestire e fare riferimento all'immagine Docker.

Per ulteriori informazioni sul refactoring dell'immagine Docker del connettore Tractus-X, consulta i grafici Refactor Tractus-X EDC Helm.

Per motivi di semplicità, evitiamo di ricostruire l'immagine del connettore secondo questo schema e utilizziamo Vault. HashiCorp

Imposta l'ordinamento specifico della lingua per i risultati delle query di Amazon Redshift utilizzando un UDF scalare in Python

Ethan Stark, Amazon Web Services

Riepilogo

Questo modello fornisce passaggi e codice di esempio per l'utilizzo di un UDF scalare in Python (funzione definita dall'utente) per configurare l'ordinamento linguistico senza distinzione tra maiuscole e minuscole per i risultati delle query di Amazon Redshift. È necessario utilizzare un UDF Python scalare perché Amazon Redshift restituisce risultati basati sull'ordinamento binario UTF-8 e non supporta l'ordinamento specifico della lingua. Un UDF Python è un codice di elaborazione non SQL basato su un programma Python ed eseguito in un data warehouse. È possibile eseguire il codice UDF Python con un'istruzione SQL in una singola query. Per ulteriori informazioni, consulta il post del blog Introduzione a Python UDFs in Amazon AWS Redshift Big Data.

I dati di esempio di questo modello si basano sull'alfabeto turco a scopo dimostrativo. L'UDF scalare Python in questo modello è stato creato per rendere i risultati delle query predefiniti di Amazon Redshift conformi all'ordinamento linguistico dei caratteri in lingua turca. Per ulteriori informazioni, consulta l'esempio della lingua turca nella sezione Informazioni aggiuntive di questo modello. È possibile modificare l'UDF scalare di Python in questo modello per altri linguaggi.

Prerequisiti e limitazioni

Prerequisiti

- Cluster Amazon Redshift con database, schema e tabelle
- Utente Amazon Redshift con `CREATE TABLE` e autorizzazioni `CREATE FUNCTION`
- Python 2.7 o successivo

Limitazioni

L'ordinamento linguistico utilizzato dalle query in questo modello non fa distinzione tra maiuscole e minuscole.

Tools (Strumenti)

Servizi AWS

- Amazon Redshift è un servizio di data warehouse gestito su scala petabyte disponibile in. Cloud AWS Amazon Redshift è integrato con il tuo data lake, il che ti consente di utilizzare i tuoi dati per acquisire nuove informazioni per la tua azienda e i tuoi clienti.

Altri strumenti

- Le funzioni definite dall'utente in Python (UDFs) sono funzioni che è possibile scrivere in Python e quindi richiamare istruzioni SQL.

Epiche

Sviluppa codice per ordinare i risultati delle query in ordine linguistico

Operazione	Description	Competenze richieste

Crea una tabella per i tuoi dati di esempio.	Per creare una tabella in Amazon Redshift e inserire i dati di esempio nella tabella, utilizza le seguenti istruzioni SQL:
	`CREATE TABLE my_table (first_name varchar(30));`
	`INSERT INTO my_table (first_name) VALUES ('ali'), ('Ali'), ('ırmak'), ('IRMAK'), ('irem'), ('İREM'), ('oğuz'), ('OĞUZ'), ('ömer'), ('ÖMER'), ('sedat'), ('SEDAT'),`
		Ingegnere dei dati

Operazione	Description	Competenze richieste
	`('şule'),` **Note** I primi nomi nei dati di esempio includono caratteri speciali dell'alfabeto turco. Per ulteriori informazioni sulle considerazioni relative alla lingua turca relative a questo esempio, consultate Esempio di lingua turca nella sezione Informazioni aggiuntive di questo modello.	

Operazione	Description	Competenze richieste

Controlla l'ordinamento predefinito dei dati di esempio.	Per visualizzare l'ordinamento predefinito dei dati di esempio in Amazon Redshift, esegui la seguente query:
	`SELECT first_name FROM my_table ORDER BY first_name;`
	La query restituisce l'elenco dei nomi dalla tabella creata in precedenza:
	`first_name`
	`---------------`
	`Ali`
	`IRMAK`
	`OĞUZ`
	`SEDAT`
	`ali`
	`irem`
	`oğuz`
	`sedat`
	`ÖMER`
	`ömer`
	`İREM`
	`ırmak`
	`ŞULE`
	`şule`
	I risultati della query non sono nell'ordine corretto perché l'ordinamento binario UTF-8 predefinito non supporta l'ordinamento linguistico dei caratteri speciali turchi.
		Ingegnere dei dati

Operazione	Description	Competenze richieste
Crea una UDF Python scalare.	Per creare una UDF Python scalare, usa il seguente codice SQL: `CREATE OR REPLACE FUNCTION collate_sort (value varchar) RETURNS varchar IMMUTABLE AS $$ def sort_str(val): import string dictionary = { 'I': 'ı', 'ı': 'h~', 'İ': 'i', 'Ş': 's~', 'ş': 's~', 'Ğ': 'g~', 'ğ': 'g~', 'Ü': 'u~', 'ü': 'u~', 'Ö': 'o~', 'ö': 'o~', 'Ç': 'c~', 'ç': 'c~' } for key, value in dictionary.items(): val = val.replace(key, value) return val.lower()`	Ingegnere dei dati

Operazione	Description	Competenze richieste

	`    return sort_str( value)`
	
	`$$ LANGUAGE plpythonu;`	

Interroga i dati di esempio.	Per interrogare i dati di esempio utilizzando Python UDF, esegui la seguente query SQL:
	
	`SELECT first_name FROM my_table ORDER BY collate_sort(first _name);`
	
	La query ora restituisce i dati di esempio in ordine linguistico turco:
	
	`first_name`
	`---------------`
	`ali`
	`Ali`
	`ırmak`
	`IRMAK`
	`irem`
	`İREM`
	`oğuz`
	`OĞUZ`
	`ömer`
	`Ömer`
	`sedat`
	`SEDAT`
	`şule`
	`ŞULE`
		Ingegnere dei dati

Risorse correlate

- Clausola ORDER BY (documentazione Amazon Redshift)
- Creazione di una UDF scalare in Python (documentazione Amazon Redshift)

Informazioni aggiuntive

Esempio di lingua turca

Amazon Redshift restituisce i risultati delle query in base all'ordinamento binario UTF-8, non all'ordinamento specifico della lingua. Ciò significa che se esegui una query su una tabella Amazon Redshift contenente caratteri turchi, i risultati della query non vengono ordinati in base all'ordinamento linguistico della lingua turca. La lingua turca contiene sei caratteri speciali (ç, ı, ğ, ö, ş e ü) che non compaiono nell'alfabeto latino. Questi caratteri speciali vengono posizionati alla fine di un set di risultati ordinato in base all'ordinamento binario UTF-8, come illustrato nella tabella seguente.

Ordinamento binario UTF-8	Ordinamento linguistico turco
a	a
b	b
c	c
d	ç (*)
e	d
f	e
g	f
h	g
i	ğ (*)
j	h
k	ı (*)

l	i
m	j
n	k
o	l
p	m
r	n
s	o
t	ö (*)
u	p
v	r
y	s
z	s (*)
c (*)	t
ğ (*)	u
ı (*)	ü (*)
ö (*)	v
s (*)	y
ü (*)	z

> **Note**
>
> L'asterisco (*) indica un carattere speciale nella lingua turca.

Come illustrato nella tabella precedente, il carattere speciale ç è compreso tra c e d nell'ordinamento linguistico turco, ma appare dopo z nell'ordinamento binario UTF-8. L'UDF scalare Python in questo modello utilizza il seguente dizionario di sostituzione dei caratteri per sostituire i caratteri speciali turchi con i corrispondenti caratteri equivalenti al latino.

Carattere speciale turco	Carattere equivalente al latino
ç	c~
ı	h ~
ğ	g ~
ö	o~
ş	s~
ü	u~

> **Note**
>
> Un carattere tilde (~) viene aggiunto alla fine dei caratteri latini che sostituiscono i corrispondenti caratteri speciali turchi.

Modifica una funzione UDF Python scalare

Per modificare la funzione UDF scalare di Python da questo modello in modo che la funzione accetti un parametro locate e supporti un dizionario di transazioni multiple, usa il seguente codice SQL:

```
CREATE OR REPLACE FUNCTION collate_sort (value varchar, locale varchar)
RETURNS varchar
IMMUTABLE
AS
$$
    def sort_str(val):
       import string
       # Turkish Dictionary
       if locale == 'tr-TR':
            dictionary = {
```

```
                'I': 'ı',
                'ı': 'h~',
                'İ': 'i',
                'Ş': 's~',
                'ş': 's~',
                'Ğ': 'g~',
                'ğ': 'g~',
                'Ü': 'u~',
                'ü': 'u~',
                'Ö': 'o~',
                'ö': 'o~',
                'Ç': 'c~',
                'ç': 'c~'
        }
        # German Dictionary
        if locale == 'de-DE':
            dictionary = {
                ....
                ....
        }

        for key, value in dictionary.items():
            val = val.replace(key, value)

        return val.lower()

    return sort_str(value)

$$ LANGUAGE plpythonu;
```

Il codice di esempio seguente mostra come interrogare l'UDF Python modificato:

```
SELECT first_name FROM my_table ORDER BY collate_order(first_name, 'tr-TR');
```

Sottoscrivi una funzione Lambda alle notifiche di eventi dai bucket S3 in diverse regioni AWS

Suresh Konathala, Amazon Web Services

Andrew Preston e Arindom Sarkar, Nessuno

Riepilogo

Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) Event Notifications pubblica notifiche per determinati eventi nel tuo bucket S3 (ad esempio, eventi di creazione di oggetti, eventi di rimozione di oggetti o eventi di ripristino di oggetti). Puoi utilizzare una funzione AWS Lambda per elaborare queste notifiche in base ai requisiti dell'applicazione. Tuttavia, la funzione Lambda non può sottoscrivere direttamente le notifiche dei bucket S3 ospitati in diverse regioni AWS.

L'approccio di questo pattern implementa uno scenario di fanout per elaborare le notifiche Amazon S3 da bucket S3 interregionali utilizzando un argomento Amazon Simple Notification Service (Amazon SNS) per ogni regione. Questi argomenti SNS regionali inviano le notifiche degli eventi di Amazon S3 a una coda Amazon Simple Queue Service (Amazon SQS) in una regione centrale che contiene anche la tua funzione Lambda. La funzione Lambda sottoscrive questa coda SQS ed elabora le notifiche degli eventi in base ai requisiti dell'organizzazione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Bucket S3 esistenti in più regioni, inclusa una regione centrale per ospitare la coda Amazon SQS e la funzione Lambda.
- AWS Command Line Interface (AWS CLI), installata e configurata. Per ulteriori informazioni su questo argomento, consulta Installazione, aggiornamento e disinstallazione dell'interfaccia a riga di comando di AWS nella documentazione dell'interfaccia a riga di comando di AWS.
- Familiarità con lo scenario fanout in Amazon SNS. Per ulteriori informazioni su questo argomento, consulta gli scenari comuni di Amazon SNS nella documentazione di Amazon SNS.

Architecture

Il diagramma seguente mostra l'architettura per l'approccio di questo pattern.

Il diagramma mostra il flusso di lavoro seguente:

1. Amazon S3 invia notifiche di eventi sui bucket S3 (ad esempio, oggetti creati, oggetti rimossi o oggetti ripristinati) a un argomento SNS nella stessa regione.
2. L'argomento SNS pubblica l'evento in una coda SQS nella regione centrale.
3. La coda SQS è configurata come origine degli eventi per la funzione Lambda e memorizza nel buffer i messaggi di evento per la funzione Lambda.
4. La funzione Lambda analizza la coda SQS alla ricerca di messaggi ed elabora le notifiche degli eventi di Amazon S3 in base ai requisiti dell'applicazione.

Stack tecnologico

- Lambda
- Amazon SNS
- Amazon SQS
- Simple Storage Service (Amazon S3)

Tools (Strumenti)

- AWS CLI — L'AWS Command Line Interface (AWS CLI) è uno strumento open source per interagire con i servizi AWS tramite comandi nella shell della riga di comando. Con una configurazione minima, puoi eseguire comandi AWS CLI che implementano funzionalità equivalenti a quelle fornite dalla Console di gestione AWS basata su browser da un prompt dei comandi.
- AWS CloudFormation: AWS ti CloudFormation aiuta a modellare e configurare le tue risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle per tutto il loro ciclo di vita. Puoi utilizzare un modello per descrivere le tue risorse e le loro dipendenze e lanciarle e configurarle insieme come uno stack, invece di gestire le risorse singolarmente. Puoi gestire e fornire stack su più account AWS e regioni AWS.
- AWS Lambda: AWS Lambda è un servizio di elaborazione che supporta l'esecuzione di codice senza effettuare il provisioning o la gestione di server. Lambda esegue il codice solo quando è

necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo. Verrà addebitato soltanto il tempo di calcolo consumato e non verrà addebitato alcun costo quando il codice non è in esecuzione.

- Amazon SNS — Amazon Simple Notification Service (Amazon SNS) coordina e gestisce la consegna o l'invio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail. I sottoscrittori ricevono tutti gli stessi messaggi pubblicati sugli argomenti ai quali sono hanno effettuato la sottoscrizione.
- Amazon SQS — Amazon Simple Queue Service (Amazon SQS) offre una coda ospitata sicura, durevole e disponibile che consente di integrare e disaccoppiare sistemi e componenti software distribuiti. Amazon SQS supporta code standard e FIFO.

Epiche

Crea la coda SQS e la funzione Lambda nella tua regione centrale

Operazione	Description	Competenze richieste
Crea una coda SQS con un trigger Lambda.	Accedi alla Console di gestione AWS e utilizza le istruzioni del tutorial Using Lambda with Amazon SQS nella documentazione di AWS Lambda per creare le seguenti risorse nella tua regione centrale: • Un ruolo di esecuzione Lambda • Una funzione Lambda per elaborare gli eventi Amazon S3 • Una coda SQS	AWS DevOps, architetto cloud

Operazione	Description	Competenze richieste
	Note Assicurati di configurare la coda SQS come origine degli eventi per la tua funzione Lambda.	

Crea un argomento SNS e configura le notifiche degli eventi per i bucket S3 in ogni regione richiesta

Operazione	Description	Competenze richieste
Crea un argomento SNS per ricevere notifiche sugli eventi di Amazon S3.	Crea un argomento SNS in una regione da cui desideri ricevere notifiche sugli eventi di Amazon S3. Per ulteriori informazioni su questo argomento, consulta l'argomento Creazione di un SNS nella documentazione di Amazon SNS. **Important** Assicurati di registrare l'Amazon Resource Name (ARN) del tuo argomento SNS.	AWS DevOps, architetto cloud
Sottoscrivi l'argomento SNS alla coda SQS centrale.	Sottoscrivi il tuo argomento SNS alla coda SQS ospitata dalla tua regione centrale. Per ulteriori informazioni su questo argomento, consulta	AWS DevOps, architetto cloud

Operazione	Description	Competenze richieste
	l'argomento Abbonamento a un SNS nella documentazione di Amazon SNS.	

Operazione	Description	Competenze richieste
Aggiorna la politica di accesso dell'argomento SNS.	1. Apri la console Amazon SNS, scegli Argomenti , quindi scegli l'argomento SNS che hai creato in precedenza. 2. Scegli Modifica, quindi espandi la sezione Politica di accesso - opzionale. 3. Allega la seguente politica di accesso al tuo argomento SNS per consentire **`sns:publish`** l'autorizzazione per Amazon S3, quindi scegli Salva: `{ "Version": "2012-10-17", "Statement": [ { "Sid": "0", "Effect": "Allow", "Principal": { "Service": "s3.amazonaws.com" }, "Action": "sns:Publish", "Resource": "arn:aws:sns:us-west-2::s3Events-SNSTopic-us-west-2" } ] }`	AWS DevOps, architetto cloud

Operazione	Description	Competenze richieste
Configura le notifiche per ogni bucket S3 nella regione.	Imposta le notifiche degli eventi per ogni bucket S3 nella regione. Per ulteriori informazioni su questo argomento, consulta Attivazione e configurazione delle notifiche di eventi utilizzando la console Amazon S3 nella documentazione di Amazon S3. **Note** Nella sezione Destinazione, scegli l'argomento SNS e specifica l'ARN dell'argomento SNS creato in precedenza.	AWS DevOps, architetto cloud
Ripeti questa storia epica per tutte le regioni richieste.	**Important** Ripeti le attività di questa epopea per ogni regione da cui desideri ricevere notifiche sugli eventi di Amazon S3, inclusa la tua regione centrale.	AWS DevOps, architetto cloud

Risorse correlate

- Configurazione di una policy di accesso (documentazione Amazon SQS)
- Configurazione di una coda SQS come origine di eventi (documentazione AWS Lambda)

- Configurazione di una coda SQS per avviare una funzione Lambda (documentazione Amazon SQS)
- AWS::Lambda::Function risorsa (CloudFormation documentazione AWS)

Tre tipi di job ETL di AWS Glue per la conversione dei dati in Apache Parquet

Adnan Alvee, Nith Govindasivan e Karthikeyan Ramachandran, Amazon Web Services

Riepilogo

Sul cloud Amazon Web Services (AWS), AWS Glue è un servizio di estrazione, trasformazione e caricamento (ETL) completamente gestito. AWS Glue rende conveniente classificare i dati, pulirli, arricchirli e spostarli in modo affidabile tra vari archivi e flussi di dati.

Questo modello fornisce diversi tipi di lavoro in AWS Glue e utilizza tre diversi script per dimostrare la creazione di lavori ETL.

Puoi usare AWS Glue per scrivere lavori ETL in un ambiente shell Python. Puoi anche creare lavori ETL in batch e in streaming utilizzando Python PySpark () o Scala in un ambiente Apache Spark gestito. Per iniziare a creare lavori ETL, questo modello si concentra sui lavori ETL in batch che utilizzano la shell Python e Scala. PySpark I job in Python shell sono pensati per carichi di lavoro che richiedono una potenza di calcolo inferiore. L'ambiente Apache Spark gestito è pensato per carichi di lavoro che richiedono un'elevata potenza di calcolo.

Apache Parquet è progettato per supportare schemi di compressione e codifica efficienti. Può velocizzare i carichi di lavoro di analisi perché archivia i dati in modo colonnare. La conversione dei dati in Parquet può far risparmiare spazio, costi e tempo di archiviazione a lungo termine. Per saperne di più su Parquet, consulta il post del blog Apache Parquet: How to be a hero with the open source columnar data format.

Prerequisiti e limitazioni

Prerequisiti

- Ruolo AWS Identity and Access Management (IAM) (se non hai un ruolo, consulta la sezione Informazioni aggiuntive).

Architecture

Stack tecnologico Target

- AWS Glue
- Amazon Simple Storage Service (Amazon S3)
- Apache Parquet

Automazione e scalabilità

- I flussi di lavoro AWS Glue supportano l'automazione completa di una pipeline ETL.
- Puoi modificare il numero di unità di elaborazione dati (DPUs), o i tipi di lavoratori, in modo da scalare orizzontalmente e verticalmente.

Tools (Strumenti)

Servizi AWS

- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Glue è un servizio ETL completamente gestito per la categorizzazione, la pulizia, l'arricchimento e lo spostamento dei dati tra vari archivi e flussi di dati.

Altri strumenti

- Apache Parquet è un formato di file di dati open source orientato alle colonne progettato per l'archiviazione e il recupero.

Configurazione

Utilizza le seguenti impostazioni per configurare la potenza di calcolo di AWS Glue ETL. Per ridurre i costi, utilizza le impostazioni minime quando esegui il carico di lavoro fornito in questo schema.

- Shell Python: è possibile utilizzare 1 DPU per utilizzare 16 GB di memoria o 0,0625 DPU per utilizzare 1 GB di memoria. Questo modello utilizza 0,0625 DPU, che è l'impostazione predefinita nella console AWS Glue.
- Python o Scala per Spark: se scegli i tipi di lavoro relativi a Spark nella console, AWS Glue per impostazione predefinita utilizza 10 worker e il tipo di worker G.1X. Questo modello utilizza due lavoratori, che è il numero minimo consentito, con il tipo di lavoratore standard, che è sufficiente ed economico.

La tabella seguente mostra i diversi tipi di worker AWS Glue per l'ambiente Apache Spark. Poiché un job della shell Python non utilizza l'ambiente Apache Spark per eseguire Python, non è incluso nella tabella.

	Standard	G.1X	G.2X
VPCU	4	4	8
Memoria	16 GB	16 GB	32 GB
Spazio su disco	50 GB	64 GB	128 GB
Esecutore per lavoratore	2	1	1

Codice

Per il codice utilizzato in questo modello, inclusa la configurazione del ruolo e dei parametri IAM, consulta la sezione Informazioni aggiuntive.

Epiche

Carica i dati

Operazione	Description	Competenze richieste
Carica i dati in un bucket S3 nuovo o esistente.	Crea o usa un bucket S3 esistente nel tuo account. Carica il file sample_data.csv dalla sezione Allegati e annota la posizione del bucket S3 e del prefisso.	Informazioni generali su AWS

Crea ed esegui il job AWS Glue

Operazione	Description	Competenze richieste
Crea il job AWS Glue.	Nella sezione ETL della console AWS Glue, aggiungi un job AWS Glue. Seleziona il tipo di lavoro appropriato, la versione di AWS Glue e il DPU/Worker tipo e il numero di lavoratori corrispondenti. Per i dettagli, consulta la sezione Configurazione.	Sviluppatore, cloud o dati
Modifica le posizioni di input e output.	Copia il codice corrispon dente al tuo job AWS Glue e modifica la posizione di input e output che hai annotato nell'epopea Upload the data.	Sviluppatore, cloud o dati
Configura i parametri.	È possibile utilizzare gli snippet forniti nella sezione Informazioni aggiuntive per impostare i parametri per il job ETL. AWS Glue utilizza internamente quattro nomi di argomenti: • `--conf` • `--debug` • `--mode` • `--JOB_NAME` Il `--JOB_NAME` parametro deve essere inserito in modo esplicito nella console AWS Glue. Scegliete Jobs, Edit	Sviluppatore, cloud o dati

Operazione	Description	Competenze richieste
	Job, Security configuration, librerie di script e parametri del lavoro (opzionale). Immettete `--JOB_NAME` come chiave e fornite un valore. Puoi anche utilizzare l'AWS Command Line Interface (AWS CLI) o l'API AWS Glue per impostare questo parametro. Il `--JOB_NAME` parametro è usato da Spark e non è necessario in un job in ambiente shell Python. È necessario aggiungere `--` prima il nome di ogni parametro; in caso contrario , il codice non funzionerà. Ad esempio, per i frammenti di codice, i parametri di posizione devono essere richiamati da and. `--input_loc --output_loc`	
Esegui il job ETL.	Esegui il tuo lavoro e controlla l'output. Nota quanto spazio è stato ridotto rispetto al file originale.	Sviluppatore, cloud o dati

Risorse correlate

Riferimenti

- Apache Spark
- AWS Glue: come funziona

- Prezzi di AWS Glue

Tutorial e video

- Cos'è AWS Glue?

Informazioni aggiuntive

Ruolo IAM

Quando crei i job AWS Glue, puoi utilizzare un ruolo IAM esistente con le autorizzazioni mostrate nel seguente frammento di codice o un nuovo ruolo.

Per creare un nuovo ruolo, usa il seguente codice YAML.

```
# (c) 2022 Amazon Web Services, Inc. or its affiliates. All Rights Reserved. This AWS
 Content is provided subject to the terms of the AWS Customer
# Agreement available at https://aws.amazon.com/agreement/ or other written agreement
 between Customer and Amazon Web Services, Inc.

AWSTemplateFormatVersion: "2010-09-09"

Description: This template will setup IAM role for AWS Glue service.

Resources:
  rGlueRole:
    Type: AWS::IAM::Role
    Properties:
      AssumeRolePolicyDocument:
        Version: "2012-10-17"
        Statement:
          - Effect: "Allow"
            Principal:
              Service:
                - "glue.amazonaws.com"
            Action:
              - "sts:AssumeRole"
      ManagedPolicyArns:
        - arn:aws:iam::aws:policy/service-role/AWSGlueServiceRole
      Policies:
        - PolicyName: !Sub "${AWS::StackName}-s3-limited-read-write-inline-policy"
          PolicyDocument:
```

```
            Version: "2012-10-17"
            Statement:
              - Effect: Allow
                Action:
                  - "s3:PutObject"
                  - "s3:GetObject"
                Resource: "arn:aws:s3:::*/*"
      Tags:
        - Key  : "Name"
          Value : !Sub "${AWS::StackName}"

Outputs:
  oGlueRoleName:
    Description: AWS Glue IAM role
    Value:
      Ref: rGlueRole
    Export:
      Name: !Join [ ":", [ !Ref "AWS::StackName", rGlueRole ] ]
```

Shell Python di AWS Glue

Il codice Python utilizza Pandas e le PyArrow librerie per convertire i dati in Parquet. La libreria Pandas è già disponibile. La PyArrow libreria viene scaricata quando si esegue il pattern, perché viene eseguita una sola volta. È possibile utilizzare i file wheel PyArrow per convertirli in una libreria e fornire il file come pacchetto di libreria. Per ulteriori informazioni sulla creazione di pacchetti di file wheel, consultate Fornire la propria libreria Python.

Parametri della shell AWS Glue Python

```
from awsglue.utils import getResolvedOptions

args = getResolvedOptions(sys.argv, ["input_loc", "output_loc"])
```

Codice shell AWS Glue Python

```
from io import BytesIO
import pandas as pd
import boto3
import os
import io
import site
from importlib import reload
```

```
from setuptools.command import easy_install
install_path = os.environ['GLUE_INSTALLATION']
easy_install.main( ["--install-dir", install_path, "pyarrow"] )
reload(site)
import pyarrow


input_loc = "s3://bucket-name/prefix/sample_data.csv"
output_loc = "s3://bucket-name/prefix/"


input_bucket = input_loc.split('/', 1)[0]
object_key = input_loc.split('/', 1)[1]


output_loc_bucket = output_loc.split('/', 1)[0]
output_loc_prefix = output_loc.split('/', 1)[1]


s3 = boto3.client('s3')
obj = s3.get_object(Bucket=input_bucket, Key=object_key)
df = pd.read_csv(io.BytesIO(obj['Body'].read()))


parquet_buffer = BytesIO()
s3_resource = boto3.resource('s3')
df.to_parquet(parquet_buffer, index=False)
s3_resource.Object(output_loc_bucket, output_loc_prefix +  'data' +
 '.parquet').put(Body=parquet_buffer.getvalue())
```

Processo AWS Glue Spark con Python

Per utilizzare un tipo di lavoro AWS Glue Spark con Python, scegli Spark come tipo di lavoro. Scegli Spark 3.1, Python 3 con tempi di avvio del processo migliorati (Glue versione 3.0) come versione AWS Glue.

Parametri di AWS Glue Python

```
from awsglue.utils import getResolvedOptions

args = getResolvedOptions(sys.argv, ["JOB_NAME", "input_loc", "output_loc"])
```

Processo AWS Glue Spark con codice Python

```
import sys
from pyspark.context import SparkContext
from awsglue.context import GlueContext
from awsglue.transforms import *
from awsglue.dynamicframe import DynamicFrame
from awsglue.utils import getResolvedOptions
from awsglue.job import Job


sc = SparkContext()
glueContext = GlueContext(sc)
spark = glueContext.spark_session
job = Job(glueContext)

input_loc = "s3://bucket-name/prefix/sample_data.csv"
output_loc = "s3://bucket-name/prefix/"

inputDyf = glueContext.create_dynamic_frame_from_options(\
    connection_type = "s3", \
    connection_options = {
        "paths": [input_loc]}, \
    format = "csv",
    format_options={
        "withHeader": True,
        "separator": ","
    })


outputDF = glueContext.write_dynamic_frame.from_options(\
    frame = inputDyf, \
    connection_type = "s3", \
    connection_options = {"path": output_loc \
        }, format = "parquet")
```

Per un gran numero di file compressi di grandi dimensioni (ad esempio, 1.000 file di circa 3 MB ciascuno), usa il `compressionType` parametro con il `recurse` parametro per leggere tutti i file disponibili all'interno del prefisso, come mostrato nel codice seguente.

```
input_loc = "bucket-name/prefix/"
output_loc = "bucket-name/prefix/"

inputDyf = glueContext.create_dynamic_frame_from_options(
```

```
                    connection_type = "s3",
                    connection_options = {"paths": [input_loc],
                                          "compressionType":"gzip","recurse" :"True",
                                          },
                    format = "csv",
                    format_options={"withHeader": True,"separator": ","}
                    )
```

Per un numero elevato di file compressi di piccole dimensioni (ad esempio 1.000 file di circa 133 KB ciascuno), utilizzate il `groupFiles` parametro insieme ai parametri `compressionType` e. `recurse` Il `groupFiles` parametro raggruppa file di piccole dimensioni in più file di grandi dimensioni e controlla il raggruppamento alla dimensione specificata in byte (ad esempio, 1 MB). `groupSize` Il seguente frammento di codice fornisce un esempio di utilizzo di questi parametri all'interno del codice.

```
input_loc = "bucket-name/prefix/"
output_loc = "bucket-name/prefix/"

inputDyf = glueContext.create_dynamic_frame_from_options(
                    connection_type = "s3",
                    connection_options = {"paths": [input_loc],
                                          "compressionType":"gzip","recurse" :"True",
                                           "groupFiles" :"inPartition",
 "groupSize" :"1048576",
                                          },
                    format = "csv",
                    format_options={"withHeader": True,"separator": ","}
                    )
```

Senza alcuna modifica nei nodi di lavoro, queste impostazioni consentono al job AWS Glue di leggere più file (grandi o piccoli, con o senza compressione) e di scriverli sulla destinazione in formato Parquet.

Lavoro in AWS Glue Spark con Scala

Per utilizzare un tipo di lavoro AWS Glue Spark con Scala, scegli Spark come tipo di lavoro e Language come Scala. Scegli Spark 3.1, Scala 2 con tempi di avvio del lavoro migliorati (Glue versione 3.0) come versione AWS Glue. Per risparmiare spazio di archiviazione, anche il seguente esempio di AWS Glue with Scala utilizza la `applyMapping` funzionalità per convertire i tipi di dati.

Parametri di AWS Glue Scala

```
import com.amazonaws.services.glue.util.GlueArgParser val args =
 GlueArgParser.getResolvedOptions(sysArgs, Seq("JOB_NAME", "inputLoc",
 "outputLoc").toArray)
```

Job AWS Glue Spark con codice Scala

```
import com.amazonaws.services.glue.GlueContext
import com.amazonaws.services.glue.MappingSpec
import com.amazonaws.services.glue.DynamicFrame
import com.amazonaws.services.glue.errors.CallSite
import com.amazonaws.services.glue.util.GlueArgParser
import com.amazonaws.services.glue.util.Job
import com.amazonaws.services.glue.util.JsonOptions
import org.apache.spark.SparkContext
import scala.collection.JavaConverters._


object GlueScalaApp {
  def main(sysArgs: Array[String]) {

    @transient val spark: SparkContext = SparkContext.getOrCreate()
    val glueContext: GlueContext = new GlueContext(spark)

    val inputLoc = "s3://bucket-name/prefix/sample_data.csv"
    val outputLoc = "s3://bucket-name/prefix/"

    val readCSV = glueContext.getSource("csv", JsonOptions(Map("paths" ->
 Set(inputLoc)))).getDynamicFrame()

    val applyMapping = readCSV.applyMapping(mappings = Seq(("_c0", "string", "date",
 "string"), ("_c1", "string", "sales", "long"),
    ("_c2", "string", "profit", "double")), caseSensitive = false)

    val formatPartition = applyMapping.toDF().coalesce(1)

    val dynamicFrame = DynamicFrame(formatPartition, glueContext)

    val dataSink = glueContext.getSinkWithFormat(
        connectionType = "s3",
        options = JsonOptions(Map("path" -> outputLoc )),
        transformationContext = "dataSink", format =
 "parquet").writeDynamicFrame(dynamicFrame)
  }
```

```
}
```

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Visualizza i log di controllo di Amazon Redshift utilizzando Amazon Athena e Amazon QuickSight

Sanket Sirsikar e Gopal Krishna Bhatia, Amazon Web Services

Riepilogo

La sicurezza è parte integrante delle operazioni di database sul cloud Amazon Web Services (AWS). L'organizzazione deve assicurarsi di monitorare le attività e le connessioni degli utenti del database per rilevare potenziali incidenti e rischi di sicurezza. Questo modello consente di monitorare i database per scopi di sicurezza e risoluzione dei problemi, un processo noto come controllo del database.

Questo modello fornisce uno script SQL che automatizza la creazione di una tabella Amazon Athena e di viste per una dashboard di reporting in Amazon che ti aiuta a controllare i log di QuickSight Amazon Redshift. Ciò garantisce che gli utenti responsabili del monitoraggio delle attività del database abbiano un comodo accesso alle funzionalità di sicurezza dei dati.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un cluster Amazon Redshift esistente. Per ulteriori informazioni su questo argomento, consulta Creare un cluster Amazon Redshift nella documentazione di Amazon Redshift.
- Accesso a un gruppo di lavoro Athena esistente. Per ulteriori informazioni, consulta Come funzionano i gruppi di lavoro nella documentazione di Amazon Athena.
- Un bucket sorgente Amazon Simple Storage Service (Amazon S3) Simple Storage Service (IAM) esistente con le autorizzazioni AWS Identity and Access Management (IAM) richieste. Per ulteriori informazioni, consulta le autorizzazioni Bucket per la registrazione di audit di Amazon Redshift da Database audit logging nella documentazione di Amazon Redshift.

Architecture

Stack tecnologico

- Athena
- Amazon Redshift
- Simple Storage Service (Amazon S3)
- QuickSight

Tools (Strumenti)

- Amazon Athena — Athena è un servizio di query interattivo che semplifica l'analisi dei dati in Amazon S3 utilizzando SQL standard.
- Amazon QuickSight: QuickSight è un servizio di business intelligence (BI) scalabile, senza server, incorporabile e basato sull'apprendimento automatico.
- Amazon Redshift — Amazon Redshift è un servizio di data warehousing di livello aziendale, su scala petabyte, completamente gestito.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è uno storage per Internet.

Epiche

Configurazione del cluster Amazon Redshift

Operazione	Description	Competenze richieste
Abilita la registrazione di controllo per il cluster Amazon Redshift.	1. Accedi alla Console di gestione AWS, apri la console Amazon Redshift, scegli CLUSTERS, quindi scegli il cluster per cui desideri abilitare la registrazione. 2. Scegli la scheda Proprietà e abilita il controllo seguendo le istruzioni di Configurazione del controllo utilizzando la console nella	DBA, ingegnere dei dati

Operazione	Description	Competenze richieste
	documentazione di Amazon Redshift.	

Operazione	Description	Competenze richieste
Abilita la registrazione nel gruppo di parametri del cluster Amazon Redshift.	Puoi abilitare il controllo dei log di connessione, dei log degli utenti e dei log delle attività degli utenti contemporaneamente utilizzando la Console di gestione AWS, il riferimento all'API Amazon Redshift o AWS Command Line Interface (AWS CLI). Per il controllo dei log delle attività degli utenti, è necessario abilitare il parametro del database. `enable_user_activity_logging` Se si abilita solo la funzionalità di registrazione di controllo ma non il parametro associato, l'audit del database registra le informazioni di registrazione per la connessione e i registri degli utenti ma non per i registri delle attività dell'utente. Il `enable_user_activity_logging` parametro non è abilitato per impostazione predefinita, ma è possibile abilitarlo modificandolo da a. `false` `true` **Important** Devi creare un nuovo gruppo di parametri del cluster	DBA, ingegnere dei dati

Operazione	Description	Competenze richieste
	con il `user_activity_logging` parametro abilitato e collegarlo al tuo cluster Amazon Redshift. Per ulteriori informazioni su questo argomento, consulta Modificare un cluster nella documentazione di Amazon Redshift. Per ulteriori informazioni su questa attività, consulta i gruppi di parametri di Amazon Redshift e Configuring auditing using the console nella documentazione di Amazon Redshift.	

Operazione	Description	Competenze richieste
Configura le autorizzazioni del bucket S3 per la registrazione dei cluster Amazon Redshift.	Quando abiliti la registrazione, Amazon Redshift raccoglie le informazioni di registrazione e le carica in file di registro archiviati in un bucket S3. Puoi utilizzare un bucket S3 esistente o crearne uno nuovo. **Important** Assicurati che Amazon Redshift disponga delle autorizzazioni IAM necessarie per accedere al bucket S3. Per ulteriori informazioni su questo argomento, consulta le autorizzazioni Bucket per la registrazione di audit di Amazon Redshift da Database audit logging nella documentazione di Amazon Redshift.	DBA, ingegnere dei dati

Creare la tabella e le viste Athena

Operazione	Description	Competenze richieste
Crea la tabella e le viste Athena per interrogare i dati	Apri la console Amazon Athena e usa la query DDL (Data Definition Language)	Ingegnere dei dati

Operazione	Description	Competenze richieste
del log di controllo di Amazon Redshift dal bucket S3.	dallo script `AuditLogging.sql` SQL (allegato) per creare la tabella e le viste per i log delle attività degli utenti, i log degli utenti e i log delle connessioni. Per ulteriori informazioni e istruzioni, consulta il tutorial Crea tabelle ed esegui query da Amazon Athena Workshop.	

Configura il monitoraggio dei registri nella QuickSight dashboard

Operazione	Description	Competenze richieste
Crea una QuickSight dashboard utilizzando Athena come origine dati.	Apri la QuickSight console Amazon e crea una QuickSight dashboard seguendo le istruzioni nel tutorial Visualizza con Athena del QuickSight Workshop di Amazon Athena.	DBA, ingegnere dei dati

Risorse correlate

- Crea tabelle ed esegui interrogazioni in Athena
- Visualizza QuickSight con Athena

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Visualizza i report sulle credenziali IAM per tutti gli account AWS utilizzando Amazon Quick Sight

Parag Nagwekar e Arun Chandapillai, Amazon Web Services

Riepilogo

Avviso: gli utenti IAM dispongono di credenziali a lungo termine, il che rappresenta un rischio per la sicurezza. Per ridurre questo rischio, si consiglia di fornire a questi utenti solo le autorizzazioni necessarie per eseguire l'attività e di rimuoverli quando non sono più necessari.

Puoi utilizzare i report sulle credenziali di AWS Identity and Access Management (IAM) per aiutarti a soddisfare i requisiti di sicurezza, controllo e conformità della tua organizzazione. I report sulle credenziali forniscono un elenco di tutti gli utenti dei tuoi account AWS e mostrano lo stato delle loro credenziali, come password, chiavi di accesso e dispositivi di autenticazione a più fattori (MFA). Puoi utilizzare i report delle credenziali per più account AWS gestiti da AWS Organizations.

Questo modello include passaggi e codice per aiutarti a creare e condividere report sulle credenziali IAM per tutti gli account AWS della tua organizzazione utilizzando i dashboard di Amazon Quick Sight. Puoi condividere le dashboard con le parti interessate della tua organizzazione. I report possono aiutare l'organizzazione a raggiungere i seguenti risultati aziendali mirati:

- Identifica gli incidenti di sicurezza relativi agli utenti IAM
- Tieni traccia della migrazione in tempo reale degli utenti IAM all'autenticazione Single Sign-On (SSO)
- Tieni traccia delle regioni AWS a cui accedono gli utenti IAM
- Rimani conforme
- Condividi le informazioni con altre parti interessate

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo

- Un'organizzazione con account membri
- Un ruolo IAM con autorizzazioni per accedere agli account in Organizations
- AWS Command Line Interface (AWS CLI) versione 2, installata e configurata
- Un abbonamento all'edizione Amazon Quick Enterprise

Architecture

Stack tecnologico

- Amazon Athena
- Amazon EventBridge
- Amazon Quick Sight
- Amazon Simple Storage Service (Amazon S3)
- AWS Glue
- AWS Identity and Access Management (IAM)
- AWS Lambda
- AWS Organizations

Architettura Target

Il diagramma seguente mostra un'architettura per la configurazione di un flusso di lavoro che acquisisce i dati dei report sulle credenziali IAM da più account AWS.

1. EventBridge richiama una funzione Lambda ogni giorno.
2. La funzione Lambda assume un ruolo IAM in ogni account AWS dell'organizzazione. Quindi, la funzione crea il report sulle credenziali IAM e archivia i dati del report in un bucket S3 centralizzato. È necessario abilitare la crittografia e disattivare l'accesso pubblico sul bucket S3.
3. Un crawler AWS Glue esegue quotidianamente la scansione del bucket S3 e aggiorna di conseguenza la tabella Athena.
4. Quick Sight importa e analizza i dati del report sulle credenziali e crea una dashboard che può essere visualizzata e condivisa con le parti interessate.

Tools (Strumenti)

Servizi AWS

- Amazon Athena è un servizio di query interattivo che semplifica l'analisi dei dati in Amazon S3 utilizzando SQL standard.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, funzioni Lambda, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altri account AWS.
- Amazon Quick Sight è un servizio di business intelligence (BI) su scala cloud che ti aiuta a visualizzare, analizzare e generare report sui dati in un'unica dashboard. Quick Sight è un componente fondamentale di Amazon Quick, che fornisce visualizzazione interattiva dei dati, analisi in memoria SPICE, analisi integrate e condivisione di dashboard.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.

Codice

Il codice per questo pattern è disponibile nel repository. GitHub getiamcredsreport-allaccounts-org Puoi utilizzare il codice di questo repository per creare report sulle credenziali IAM su account AWS in Organizations e archiviarli in una posizione centrale.

Epiche

Configura l'infrastruttura

Operazione	Description	Competenze richieste
Configura l'edizione Amazon Quick Enterprise.	1. Attiva l'edizione Amazon Quick Enterprise nel tuo account AWS. Per ulteriori informazioni, consulta	Amministratore AWS DevOps, AWS, amministratore cloud, architetto cloud

Operazione	Description	Competenze richieste
	Gestire l'accesso degli utenti all'interno di Amazon Quick nella documentazione di Quick. 2. Per concedere le autorizzazioni del dashboard, ottieni l'Amazon Resource Name (ARN) degli utenti Quick Sight.	
Integra Amazon Quick Sight con Amazon S3 e Athena.	È necessario autorizzare Quick Sight a utilizzare Amazon S3 e Athena prima di distribuire lo stack AWS. CloudFormation	Amministratore AWS DevOps, AWS, amministratore cloud, architetto cloud

Implementa l'infrastruttura

Operazione	Description	Competenze richieste
Clona il GitHub repository.	1. Clona il GitHub getiamcredsreport-allaccounts-orgrepository sul tuo computer locale eseguendo il seguente comando: `git clone https://github.com/aws-samples/getiamcredsreport-allaccounts-org`	Amministratore AWS
Implementa l'infrastruttura.	1. Accedi alla console di gestione AWS e apri la console CloudFormation .	Amministratore AWS

Operazione	Description	Competenze richieste
	2. Nel riquadro di navigazione, scegli Crea stack, quindi scegli Con nuove risorse (standard). 3. Nella pagina Identifica risorse, scegli Avanti. 4. Nella pagina Specificare il modello, per Origine del modello, seleziona Carica un file modello. 5. Scegli file, seleziona il **`Cloudformation-createcredrepo.yaml`** file dal tuo GitHub repository clonato, quindi scegli Avanti. 6. In Parametri, esegui l'aggiornamento `IAMRoleName` con il tuo ruolo IAM. Questo dovrebbe essere il ruolo IAM che vuoi che Lambda assuma in ogni account dell'organizzazione. Questo ruolo crea il rapporto sulle credenziali. Nota: il ruolo non deve essere presente in tutti gli account in questa fase della creazione dello stack. 7. In Parametri, aggiorna `S3BucketName` con il nome del bucket S3 in cui Lambda può memorizza	

Operazione	Description	Competenze richieste
	re le credenziali per tutti gli account. 8. Per il nome dello stack, inserisci il nome dello stack. 9. Seleziona Invia. 10. Nota il nome del ruolo della funzione Lambda.	
Crea una politica di autorizzazione IAM.	Crea una policy IAM per ogni account AWS della tua organizzazione con le seguenti autorizzazioni: `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [ "iam:GenerateCredentialReport", "iam:GetCredentialReport" ], "Resource": "*" } ] }`	AWS DevOps, amministratore del cloud, architetto del cloud, ingegnere dei dati

Operazione	Description	Competenze richieste
Crea un ruolo IAM con una policy di fiducia.	1. Crea un ruolo IAM per gli account AWS e allega la policy di autorizzazione creata nel passaggio precedente. 2. Allega la seguente policy di fiducia al ruolo IAM: `{ "Version": "2012-10-17", "Statement":[ { "Effect": "Allow", "Principal":{ "AWS":[ "arn:aws:iam::<MasterAccountID>:role/<LambdaRole>" ] }, "Action": "sts:AssumeRole" } ] }` **Important** Sostituisci `arn:aws:iam::<MasterAccountID>:role/<LambdaRole>` con l'ARN del ruolo	Amministratore cloud, architetto cloud, amministratore AWS

Operazione	Description	Competenze richieste
	Lambda indicato in precedenza. **Note** Le organizzazioni in genere utilizzano l'automazione per creare ruoli IAM per i propri account AWS. Ti consigliamo di utilizzare questa automazione, se disponibile. In alternativa, puoi utilizzare lo `CreateRoleforOrg.py` script dal repository del codice. Lo script richiede un ruolo amministrativo esistente o qualsiasi altro ruolo IAM che disponga dell'autorizzazione a creare una policy e un ruolo IAM in ogni account AWS.	

Operazione	Description	Competenze richieste
Configura Amazon Quick Sight per visualizzare i dati.	1. Accedi a Quick con le tue credenziali. 2. Crea un set di dati utilizzando Athena (utilizzando `iamcredreportdb` il database `"cfn_iamcredreport"` e la tabella), quindi aggiorna automaticamente il set di dati. 3. Crea un'analisi in Quick Sight. 4. Crea una dashboard Quick Sight.	AWS DevOps, amministratore del cloud, architetto del cloud, ingegnere dei dati

Informazioni aggiuntive

Considerazioni aggiuntive

Considera i seguenti aspetti:

- Dopo aver distribuito l' CloudFormation infrastruttura, puoi attendere che Lambda e AWS Glue vengano eseguiti secondo le rispettive pianificazioni, prima che i report vengano creati in Amazon S3 e analizzati da Athena. In alternativa, puoi eseguire Lambda manualmente per ottenere i report in Amazon S3, quindi eseguire il crawler AWS Glue per ottenere la tabella Athena creata dai dati.
- Quick è un potente strumento per analizzare e visualizzare i dati in base ai requisiti aziendali. Puoi utilizzare i parametri in Quick per controllare i dati dei widget in base ai campi di dati che scegli. Inoltre, puoi utilizzare un'analisi rapida per creare parametri (ad esempio, campi Account, Data e Utente, ad esempio `partition_0`, `partition_1`, e `user` rispettivamente) dal set di dati per aggiungere controlli per i parametri Account, Data e Utente.
- Per creare dashboard Quick Sight personalizzate, consulta Quick Workshops dal sito Web di AWS Workshop Studio.

- Per vedere esempi di dashboard Quick Sight, consulta il repository di GitHub getiamcredsreport-allaccounts-orgcodice.

Risultati aziendali mirati

È possibile utilizzare questo modello per ottenere i seguenti risultati aziendali mirati:

- Identifica gli incidenti di sicurezza relativi agli utenti IAM: esamina ogni utente di ogni account AWS della tua organizzazione utilizzando un unico pannello di controllo. Puoi monitorare l'andamento delle singole regioni AWS a cui un utente IAM ha effettuato l'accesso più recente e dei servizi che ha utilizzato.
- Tieni traccia della migrazione in tempo reale degli utenti IAM all'autenticazione SSO: utilizzando SSO, gli utenti possono accedere una sola volta con una singola credenziale e accedere a più account e applicazioni AWS. Se hai intenzione di migrare i tuoi utenti IAM a SSO, questo modello può aiutarti a passare all'SSO e tenere traccia di tutto l'utilizzo delle credenziali degli utenti IAM (come l'accesso alla Console di gestione AWS o l'uso delle chiavi di accesso) su tutti gli account AWS.
- Tieni traccia delle regioni AWS a cui accedono gli utenti IAM: puoi controllare l'accesso degli utenti IAM alle regioni per vari scopi, come la sovranità dei dati e il controllo dei costi. Puoi anche tenere traccia dell'uso delle regioni da parte di qualsiasi utente IAM.
- Resta conforme: seguendo il principio del privilegio minimo, puoi concedere solo le autorizzazioni IAM specifiche necessarie per eseguire un'attività specifica. Inoltre, puoi monitorare l'accesso ai servizi AWS, alla Console di gestione AWS e l'utilizzo delle credenziali a lungo termine.
- Condividi informazioni con altre parti interessate: puoi condividere dashboard curate con altre parti interessate, senza concedere loro l'accesso ai report sulle credenziali IAM o agli account AWS.

Altri modelli

- Autentica gli utenti delle applicazioni React esistenti utilizzando Amazon Cognito e l'interfaccia utente AWS Amplify
- Estrai automaticamente i contenuti dai file PDF utilizzando Amazon Textract
- Crea un modello di previsione con avvio a freddo utilizzando DeepAR per le serie temporali in SageMaker Amazon AI Studio Lab
- Crea report dettagliati su costi e utilizzo per i cluster Amazon EMR utilizzando AWS Cost Explorer
- Crea report dettagliati su costi e utilizzo per Amazon RDS e Amazon Aurora
- Crea report dettagliati su costi e utilizzo per i lavori AWS Glue utilizzando AWS Cost Explorer
- Distribuisci i record DynamoDB ad Amazon S3 utilizzando Kinesis Data Streams e Firehose con AWS CDK
- Implementa un job AWS Glue con una pipeline CodePipeline CI/CD AWS
- Incorpora i componenti visivi di Amazon Quick Sight nelle applicazioni Web utilizzando Amazon Cognito e l'automazione IAc
- Stima dei costi di storage per una tabella Amazon DynamoDB
- Esegui le query SQL di Amazon Redshift utilizzando Terraform
- Estrarre e interrogare gli attributi AWS IoT SiteWise dei metadati in un data lake
- Genera approfondimenti sui dati utilizzando Modernizzazione del mainframe AWS Amazon Q in Quick Sight
- Genera approfondimenti sui z/OS dati Db2 utilizzando Modernizzazione del mainframe AWS Amazon Q in Quick Sight
- Offri alle istanze di SageMaker notebook l'accesso temporaneo a un CodeCommit repository in un altro account AWS
- Identifica e avvisa quando le risorse Amazon Data Firehose non sono crittografate con una chiave AWS KMS
- Importa la libreria psycopg2 per interagire con il tuo AWS Lambda database PostgreSQL
- Inserisci e analizza i log di sicurezza AWS in Microsoft Sentinel
- Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL in modalità SSL utilizzando AWS DMS
- Esegui la migrazione di un database Oracle ad Amazon RDS for Oracle utilizzando gli adattatori flat file GoldenGate Oracle

- Esegui la migrazione di un database Oracle ad Amazon Redshift utilizzando AWS DMS e AWS SCT
- Esegui la migrazione dei dati da un ambiente Hadoop locale ad Amazon S3 utilizzando AWS per Amazon S3 DistCp PrivateLink
- Migrazione da Couchbase Server a Couchbase Capella su AWS
- Esegui la migrazione dei carichi di lavoro Cloudera locali a Cloudera Data Platform su AWS
- Gestione del failover Multi-AZ per i cluster EMR utilizzando Application Recovery Controller
- Automatizza le blue/green implementazioni dei database globali di Amazon Aurora utilizzando i principi IaC
- Esegui il provisioning di AWS Service Catalog prodotti basati su AWS CloudFormation modelli utilizzando GitHub Actions
- Esegui test unitari per i lavori ETL in Python AWS Glue utilizzando il framework pytest
- Configura una dashboard di monitoraggio Grafana per AWS ParallelCluster
- Scarica i dati da un cluster Amazon Redshift tra più account su Amazon S3
- Visualizza i risultati AI/ML del modello utilizzando Flask e AWS Elastic Beanstalk

Calcolo

Argomenti

- Contenitori e microservizi
- Serverless
- Rete
- Distribuzione di contenuti

Contenitori e microservizi

Argomenti

- Accedi a un database Amazon Neptune da un container Amazon EKS
- Accedi alle applicazioni container in modo privato su Amazon ECS utilizzando AWS PrivateLink e un Network Load Balancer
- Accedi alle applicazioni container in modo privato su Amazon ECS utilizzando AWS Fargate, PrivateLink AWS e un Network Load Balancer
- Accedi alle applicazioni container in modo privato su Amazon EKS utilizzando AWS PrivateLink e un Network Load Balancer
- Automatizza i backup per le istanze DB di Amazon RDS for PostgreSQL utilizzando AWS Batch
- Automatizza la distribuzione di Node Termination Handler in Amazon EKS utilizzando una pipeline CI/CD
- Crea e distribuisci automaticamente un'applicazione Java su Amazon EKS utilizzando una pipeline CI/CD
- Copia le immagini dei container Amazon ECR su e Account AWS Regioni AWS
- Crea una definizione di attività Amazon ECS e monta un file system su EC2 istanze utilizzando Amazon EFS
- Implementa le funzioni Lambda con immagini dei container
- Distribuisci microservizi Java su Amazon ECS utilizzando AWS Fargate
- Distribuisci risorse e pacchetti Kubernetes utilizzando Amazon EKS e un repository di grafici Helm in Amazon S3
- Implementa un cluster CockroachDB in Amazon EKS utilizzando Terraform
- Implementa un microservizio Java di esempio su Amazon EKS ed esponi il microservizio utilizzando un Application Load Balancer
- Implementa un'applicazione basata su gRPC su un cluster Amazon EKS e accedi ad essa con un Application Load Balancer
- Distribuisci applicazioni containerizzate in AWS IoT Greengrass V2 esecuzione come contenitore Docker
- Distribuisci contenitori utilizzando Elastic Beanstalk
- Genera un indirizzo IP statico in uscita utilizzando una funzione Lambda, Amazon VPC e un'architettura serverless

- Identifica automaticamente le immagini duplicate dei container durante la migrazione a un repository Amazon ECR
- Installa SSM Agent sui nodi di lavoro Amazon EKS utilizzando Kubernetes DaemonSet
- Installa l'agente SSM e l' CloudWatch agente sui nodi di lavoro Amazon EKS utilizzando preBootstrapCommands
- Esegui la migrazione dei controller di ingresso NGINX quando abiliti la modalità automatica di Amazon EKS
- Esegui la migrazione dei carichi di lavoro dei container da Azure Red Hat OpenShift (ARO) a Servizio Red Hat OpenShift su AWS (ROSA)
- Esegui attività Amazon ECS su Amazon WorkSpaces con Amazon ECS Anywhere
- Esegui un contenitore Docker dell'API Web ASP.NET Core su un'istanza Amazon Linux EC2
- Esegui carichi di lavoro con stato con storage persistente dei dati utilizzando Amazon EFS su Amazon EKS con AWS Fargate
- Configura la scalabilità automatica basata sugli eventi in Amazon EKS utilizzando Amazon EKS Pod Identity e KEDA
- Semplifica le implementazioni di PostgreSQL su Amazon EKS utilizzando PGO
- Semplifica l'autenticazione delle applicazioni con TLS reciproco in Amazon ECS utilizzando Application Load Balancer
- Altri modelli

Accedi a un database Amazon Neptune da un container Amazon EKS

Ramakrishnan Palaninathan, Amazon Web Services

Riepilogo

Questo modello stabilisce una connessione tra Amazon Neptune, che è un database grafico completamente gestito, e Amazon Elastic Kubernetes Service (Amazon EKS), un servizio di orchestrazione di container, per accedere a un database Neptune. I cluster Neptune DB sono confinati all'interno di un cloud privato virtuale (VPC) attivo. AWS Per questo motivo, l'accesso a Neptune richiede un'attenta configurazione del VPC per abilitare la connettività.

A differenza di Amazon Relational Database Service (Amazon RDS) per PostgreSQL, Neptune non si basa sulle tipiche credenziali di accesso al database. Utilizza invece ruoli (IAM) per l'autenticazione. AWS Identity and Access Management Pertanto, la connessione a Neptune da Amazon EKS implica la configurazione di un ruolo IAM con le autorizzazioni necessarie per accedere a Neptune.

Inoltre, gli endpoint Neptune sono accessibili solo all'interno del VPC in cui risiede il cluster. Ciò significa che devi configurare le impostazioni di rete per facilitare la comunicazione tra Amazon EKS e Neptune. A seconda dei requisiti specifici e delle preferenze di rete, esistono diversi approcci alla configurazione del VPC per consentire una connettività perfetta tra Neptune e Amazon EKS. Ogni metodo offre vantaggi e considerazioni distinti, che offrono flessibilità nella progettazione dell'architettura del database per soddisfare le esigenze dell'applicazione.

Prerequisiti e limitazioni

Prerequisiti

- Installa l'ultima versione di kubectl (vedi istruzioni). Per verificare la tua versione, esegui:

```
kubectl version --short
```

- Installa l'ultima versione di eksctl (vedi istruzioni). Per verificare la tua versione, esegui:

```
eksctl info
```

- Installa l'ultima versione di AWS Command Line Interface (AWS CLI) versione 2 (vedi istruzioni). Per verificare la tua versione, esegui:

```
aws --version
```

- Crea un cluster Neptune DB (vedi istruzioni). Assicurati di stabilire comunicazioni tra il VPC del cluster e Amazon EKS tramite peering VPC o un altro metodo. AWS Transit Gateway Assicurati inoltre che lo stato del cluster sia «disponibile» e che abbia una regola in entrata sulla porta 8182 per il gruppo di sicurezza.
- Configura un provider IAM OpenID Connect (OIDC) su un cluster Amazon EKS esistente (vedi istruzioni).

Versioni del prodotto

- Amazon EKS 1,27
- Motore Amazon Neptune versione 1.3.0.0 (2023-11-15)

Architecture

Il diagramma seguente mostra la connessione tra i pod Kubernetes in un cluster Amazon EKS e Neptune per fornire l'accesso a un database Neptune.

Automazione e scalabilità

Puoi usare Amazon EKS Horizontal Pod Autoscaler per scalare questa soluzione.

Tools (Strumenti)

Servizi

- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire AWS Kubernetes senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon Neptune è un servizio di database a grafi che ti aiuta a creare ed eseguire applicazioni che funzionano con set di dati altamente connessi.

Best practice

Per le best practice, consulta Identity and Access Management nelle guide alle best practice di Amazon EKS.

Epiche

Impostazione delle variabili di ambiente

Operazione	Description	Competenze richieste
Verifica il contesto del cluster.	Prima di interagire con il cluster Amazon EKS utilizzando Helm o altri strumenti da riga di comando, devi definire variabili di ambiente che incapsulino i dettagli del cluster. Queste variabili vengono utilizzate nei comandi successivi per garantire che siano indirizzate al cluster e alle risorse corretti. Innanzitutto, verifica che stai operando nel contesto del cluster corretto. Ciò garantisce che tutti i comandi successivi vengano inviati al cluster Kubernetes previsto. Per verificare il contesto corrente, esegui il comando seguente. `kubectl config current-context`	Amministratore AWS, amministratore cloud
Definisci la `CLUSTER_NAME` variabile.	Definisci la variabile di `CLUSTER_NAME` ambiente per il tuo cluster Amazon EKS. Nel comando seguente, sostituisci il valore di esempio `us-west-2` con quello corretto Regione AWS per	Amministratore AWS, amministratore cloud

Operazione	Description	Competenze richieste
	il tuo cluster. Sostituisci il valore di esempio `eks-workshop` con il nome del cluster esistente. `export CLUSTER_NAME=$(aws eks describe-cluster --region us-west-2 --name eks-workshop --query "cluster.name" --output text)`	
Convalida l'output.	Per verificare che le variabili siano state impostate correttamente, esegui il comando seguente. `echo $CLUSTER_NAME` Verificate che l'output di questo comando corrispon da all'input specificato nel passaggio precedente.	Amministratore AWS, amministratore cloud

Crea un ruolo IAM e associalo a Kubernetes

Operazione	Description	Competenze richieste
Crea un account di servizio .	Utilizzi i ruoli IAM per gli account di servizio per mappare i tuoi account di servizio Kubernetes ai ruoli IAM, per consentire una gestione granulare delle autorizzazioni per le tue	Amministratore AWS, amministratore cloud

Operazione	Description	Competenze richieste
	applicazioni eseguite su Amazon EKS. Puoi usare eksctl per creare e associare un ruolo IAM a un account di servizio Kubernetes specifico all'interno del tuo cluster Amazon EKS. La policy AWS gestita `NeptuneFullAccess` consente l'accesso in scrittura e lettura al cluster Neptune specificato. **Important** È necessario che al cluster sia associato un endpoint OIDC prima di eseguire questi comandi. Crea un account di servizio da associare a una politica AWS gestita denominata. `NeptuneFullAccess` `eksctl create iamserviceaccount --name eks-neptune-sa --namespace default --cluster $CLUSTER_NAME --attach-policy-arn arn:aws:iam::aws:policy/NeptuneFullAccess --approve --`	

Operazione	Description	Competenze richieste

	`override-existing-serviceaccounts`
	
	dove `eks-neptune-sa` è il nome dell'account di servizio che desideri creare.
	
	Al termine, questo comando visualizza la seguente risposta:
	
	`2024-02-07 01:12:39 [#] created serviceaccount "default/eks-neptune-sa"`	

Operazione	Description	Competenze richieste

Verifica che l'account sia configurato correttamente.	Assicurati che l'account del `eks-neptune-sa` servizio sia configurato correttamente nello spazio dei nomi predefinito del cluster.
	`kubectl get sa eks-neptune-sa -o yaml`
	L'output dovrebbe essere simile al seguente:
	`apiVersion: v1`
	`kind: ServiceAccount`
	`metadata:`
	`  annotations:`
	`  eks.amazonaws.com/role-arn: arn:aws:iam::123456789123:role/eksctl-eks-workshop-addon-iamserviceaccount-d-Role1-Q35yKgdQOlmM`
	`  creationTimestamp: "2024-02-07T01:12:39Z"`
	`  labels:`
	`  app.kubernetes.io/managed-by: eksctl`
	`  name: eks-neptune-sa`
	`  namespace: default`
	`  resourceVersion: "5174750"`
	`  uid: cd6ba2f7-a0f5-40e1-a6f4-4081e0042316`
		Amministratore AWS, amministratore cloud

Operazione	Description	Competenze richieste

Verifica la connettività.	Implementa un pod di esempio chiamato `pod-util` e verifica la connettività con Neptune.
	
	`apiVersion: v1`
	`kind: Pod`
	`metadata:`
	`  name: pod-util`
	`  namespace: default`
	`spec:`
	`  serviceAccountName:`
	` eks-neptune-sa`
	`  containers:`
	`  - name: pod-util`
	`  image: public.ec`
	`r.aws/patrickc/tro`
	`ubleshoot-util`
	`  command:`
	`  - sleep`
	`  - "3600"`
	`  imagePullPolicy:`
	` IfNotPresent`
	
	`kubectl apply -f pod-`
	`util.yaml`
	
	`kubectl exec --stdin --`
	`tty pod-util -- /bin/`
	`bash`
	`bash-5.1# curl -X POST`
	` -d '{"gremlin":"g.V()`
	`.limit(1)"}' https://`
	`db-neptune-1.cluste`
	`r-xxxxxxxxxxx.us-`
	`west-2.neptune.ama`
	`zonaws.com:8182/gr`
	`emlin`
	`{"requestId":"a4964f2`
	`d-12b1-4ed3-8a14-e`
		Amministratore AWS, amministratore cloud

Operazione	Description	Competenze richieste
	`ff511431a0e","stat us":{"message":"", "code":200,"attrib utes":{"@type":"g: Map","@value":[]}} ,"result":{"data": {"@type":"g:List", "@value":[]},"meta ":{"@type":"g:Map" ,"@value":[]}}} bash-5.1# exit exit`	

Convalida l'attività di connessione

Operazione	Description	Competenze richieste
Abilita l'autenticazione del database IAM.	Per impostazione predefinita, l'autenticazione del database IAM è disabilitata quando si crea un cluster Neptune DB. È possibile abilitare o disabilitare l'autenticazione del database IAM utilizzando. Console di gestione AWS Segui i passaggi indicati nella AWS documentazione per abilitare l'autenticazione del database IAM in Neptune.	Amministratore AWS, amministratore cloud
Verifica le connessioni.	In questo passaggio, interagis ci con il `pod-util` contenito re, che è già in stato di esecuzione, per installar	Amministratore AWS, amministratore cloud

Operazione	Description	Competenze richieste

	e awscurl e verificare la connessione.
	
	1. Esegui il seguente comando per trovare il pod.
	
	`kubectl get pods`
	
	L'output dovrebbe essere simile al seguente:
	
	`NAME READY STATUS RESTARTS AGE pod-util 1/1 Running 0 50m`
	
	2. Esegui il comando seguente per installare awscurl.
	
	`kubectl exec --stdin --tty pod-util -- /bin/bash bash-5.1# pip3 install awscurl Installing collected packages: idna, configparser, configargparse, charset-normalizer, certifi, requests, awscurl Successfully installed awscurl-0.32 certifi-2024.2 charset-normalizer-3.3.2 configargparse-1.7 configparser-6.0.0 idna-3.6 requests-2.31.0 bash-5.1# awscurl`	

Operazione	Description	Competenze richieste
	https://db-neptune-1.cluster-xxxxxxxxxxxx.us-west-2.neptune.amazonaws.com:8182/status --region us-west-2 --service neptune-db {"status":"healthy","startTime":"Thu Feb 08 01:22:14 UTC 2024","dbEngineVersion":"1.3.0.0.R1","role":"writer","dfeQueryEngine":"viaQueryHint","gremlin":{"version":"tinkerpop-3.6.4"},"sparql":{"version":"sparql-1.1"},"opencypher":{"version":"Neptune-9.0.20190305-1.0"},"labMode":{"ObjectIndex":"disabled","ReadWriteConflictDetection":"enabled"},"features":{"SlowQueryLogs":"disabled","ResultCache":{"status":"disabled"},"IAMAuthentication":"enabled","Streams":"disabled","AuditLog":"disabled"},"settings":{"clusterQueryTimeoutInMs":"120000","SlowQueryLogsThreshold":"5000"}}	

risoluzione dei problemi

Problema	Soluzione
Impossibile accedere al database Neptune.	Esamina la policy IAM allegata all'account del servizio. Assicurati che consenta le azioni necessarie (ad esempio`neptune:Connec,neptune:DescribeDBInstances`) per le operazioni che desideri eseguire.

Risorse correlate

- Concedi ai carichi di lavoro Kubernetes l'accesso all' AWS utilizzo degli account di servizio Kubernetes (documentazione Amazon EKS)
- Ruoli IAM per gli account di servizio (documentazione Amazon EKS)
- Creazione di un nuovo cluster Neptune DB (documentazione Amazon Neptune)

Accedi alle applicazioni container in modo privato su Amazon ECS utilizzando AWS PrivateLink e un Network Load Balancer

Kirankumar Chandrashekar, Amazon Web Services

Riepilogo

Questo modello descrive come ospitare privatamente un'applicazione container Docker su Amazon Elastic Container Service (Amazon ECS) con un Network Load Balancer e accedere all'applicazione utilizzando AWS. PrivateLink Puoi quindi utilizzare una rete privata per accedere in modo sicuro ai servizi sul cloud Amazon Web Services (AWS). Amazon Relational Database Service (Amazon RDS) ospita il database relazionale per l'applicazione in esecuzione su Amazon ECS con alta disponibilità (HA). Amazon Elastic File System (Amazon EFS) viene utilizzato se l'applicazione richiede uno storage persistente.

Il servizio Amazon ECS che esegue le applicazioni Docker, con un Network Load Balancer sul front-end, può essere associato a un endpoint di cloud privato virtuale (VPC) per l'accesso tramite AWS. PrivateLink Questo servizio di endpoint VPC può quindi essere condiviso con altri utenti utilizzando i rispettivi VPCs endpoint VPC.

Puoi anche utilizzare AWS Fargate al posto di un gruppo Amazon Auto EC2 Scaling. Per ulteriori informazioni, consulta Accesso privato alle applicazioni container su Amazon ECS utilizzando AWS Fargate, PrivateLink AWS e un Network Load Balancer.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- AWS Command Line Interface (AWS CLI) versione 2, installata e configurata su Linux, macOS o Windows
- Docker, installato e configurato su Linux, macOS o Windows
- Un'applicazione in esecuzione su Docker

Architecture

Stack tecnologico

- Amazon CloudWatch
- Amazon Elastic Compute Cloud (Amazon EC2)
- Amazon EC2 Auto Scaling
- Amazon Elastic Container Registry (Amazon ECR)
- Amazon ECS
- Amazon RDS
- Amazon Simple Storage Service (Amazon S3)
- AWS Lambda
- AWS PrivateLink
- AWS Secrets Manager
- Application Load Balancer
- Network Load Balancer
- VPC

Automazione e scalabilità

- Puoi usare AWS CloudFormation per creare questo modello utilizzando Infrastructure as Code.

Tools (Strumenti)

- Amazon EC2 — Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS.
- Amazon EC2 Auto Scaling: Amazon EC2 Auto Scaling ti aiuta a garantire il numero corretto di istanze EC2 Amazon disponibili per gestire il carico della tua applicazione.
- Amazon ECS — Amazon Elastic Container Service (Amazon ECS) è un servizio di gestione dei container veloce e altamente scalabile che semplifica l'esecuzione, l'arresto e la gestione dei container su un cluster.
- Amazon ECR — Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container AWS gestito che è sicuro, scalabile e affidabile.
- Amazon EFS — Amazon Elastic File System (Amazon EFS) fornisce un file system NFS elastico semplice, scalabile e completamente gestito da utilizzare con i servizi cloud AWS e le risorse locali.
- AWS Lambda — Lambda è un servizio di calcolo per l'esecuzione di codice senza effettuare il provisioning o la gestione di server.

- Amazon RDS — Amazon Relational Database Service (Amazon RDS) è un servizio Web che semplifica la configurazione, il funzionamento e la scalabilità di un database relazionale nel cloud AWS.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è uno storage per Internet. È concepito per rendere più accessibili agli sviluppatori risorse informatiche su grande scala per il Web.
- AWS Secrets Manager — Secrets Manager ti aiuta a sostituire le credenziali codificate nel codice, comprese le password, fornendo una chiamata API a Secrets Manager per recuperare il segreto a livello di codice.
- Amazon VPC: Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito.
- Elastic Load Balancing: Elastic Load Balancing distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni, come istanze EC2 Amazon, contenitori e indirizzi IP, in più zone di disponibilità.
- Docker: Docker aiuta gli sviluppatori a imballare, spedire ed eseguire qualsiasi applicazione come contenitore leggero, portatile e autosufficiente.

Epiche

Crea componenti di rete

Operazione	Description	Competenze richieste
Crea un VPC.	1. Accedi alla Console di gestione AWS e apri la console Amazon VPC. Scegli Crea VPC e scegli VPC e altro ancora. 2. Inserisci un nome per il tuo VPC e scegli un intervallo di blocchi CIDR appropriato. 3. Specificate due zone di disponibilità, due sottoreti pubbliche, quattro sottoreti private. Due sottoreti	Amministratore cloud

Operazione	Description	Competenze richieste
	private sono per le attività di Amazon ECS e due sottoreti private sono per i database Amazon RDS. 4. Specificare un gateway NAT per ogni zona di disponibilità. 5. Seleziona Crea VPC.	

Crea i sistemi di bilanciamento del carico

Operazione	Description	Competenze richieste
Crea un Network Load Balancer.	1. Apri la EC2 console Amazon e scegli la regione AWS che contiene il tuo VPC. 2. In Load balancing, scegli Load balancer e scegli Create load balancer. 3. Scegli Network Load Balancer e scegli Crea. 4. Nella pagina Configure load balancer, configura Network Load Balancer e listener. Importante: assicurati di scegliere lo schema del Network Load Balancer come Interno. 5. Scegli le impostazioni di sicurezza applicabili, configura un gruppo di sicurezza e un gruppo	Amministratore cloud

Operazione	Description	Competenze richieste
	target. Scegli Istanza o IP come tipo di destinazione nella sezione Configura il routing. Assicurati di non registrare un obiettivo. 6. Dopo aver configurato tutte le impostazioni, scegli Avanti: Revisione, quindi scegli Crea.	

Operazione	Description	Competenze richieste
Crea un Application Load Balancer.	1. Sulla EC2 console Amazon, scegli la stessa regione che contiene il tuo VPC. 2. In Load balancing, scegli Load balancer e scegli Crea load balancer. 3. Scegliete Application Load Balancer e scegliete Crea. 4. **Important** Configura l'Application Load Balancer e il relativo listener. Assicurati di scegliere lo schema di Application Load Balancer come Interno. 5. Scegliete le impostazioni di sicurezza applicabili, configurate un gruppo di sicurezza e un gruppo target. Scegli Istanza o IP come tipo di destinazione nella sezione Configura il routing. Assicurati di non registrare un obiettivo. 6. Dopo aver configurato tutte le impostazioni, scegli Avanti: Revisione, quindi scegli Crea.	Amministratore cloud

Creare un file system Amazon EFS

Operazione	Description	Competenze richieste
Crea un file system Amazon EFS.	1. Apri la console Amazon EFS e scegli Crea file system. 2. Nella finestra di dialogo Crea file system, inserisci un nome per il tuo file system e scegli il tuo VPC. 3. Scegli Crea per creare il file system. 4. Configura e configura il tuo file system Amazon EFS.	Amministratore cloud
Monta gli obiettivi per le sottoreti.	1. Torna alla console Amazon EFS e scegli File system. La pagina File system mostra i file system Amazon EFS presenti nel tuo account. 2. Scegli il file system che hai creato e scegli Gestisci per visualizzare le zone di disponibilità. Per aggiunger e una destinazione di montaggio, scegli Aggiungi destinazione di montaggio e aggiungi le quattro sottoreti private che hai creato.	Amministratore cloud
Verifica che le sottoreti siano montate come destinazioni.	1. Sulla console Amazon EFS, scegli File system. 2. Scegli Rete per visualizz are l'elenco dei target	Amministratore cloud

Operazione	Description	Competenze richieste
	di montaggio esistenti. Assicurati che includano le quattro sottoreti che hai creato.	

Creare un bucket S3

Operazione	Description	Competenze richieste
Crea un bucket S3.	Apri la console Amazon S3 e crea un bucket S3 per archiviare gli asset statici dell'applicazione, se necessario.	Amministratore cloud

Crea un segreto di Secrets Manager

Operazione	Description	Competenze richieste
Crea una chiave AWS KMS per crittografare il segreto di Secrets Manager.	Apri la console AWS Key Management Service (AWS KMS) e crea una chiave KMS.	Amministratore del cloud
Crea un segreto di Secrets Manager per archiviare la password di Amazon RDS.	1. Apri la console AWS Secrets Manager e crea un nuovo segreto scegliendo Archivia un nuovo segreto. 2. Scegli la chiave KMS che hai creato e archivia il tuo nuovo segreto.	Amministratore cloud

Crea un'istanza Amazon RDS

Operazione	Description	Competenze richieste
Creazione di un gruppo di sottoreti del database.	1. Apri la console Amazon RDS e scegli Gruppi di sottorete. 2. Scegli Crea gruppo di sottoreti DB e inserisci un nome e una descrizione per il tuo gruppo di sottoreti DB. 3. Scegli il VPC che hai creato in precedenza e scegli le zone di disponibilità e le sottoreti. Quindi, scegli Crea.	Amministratore cloud
Crea un'istanza Amazon RDS.	Crea e configura un'istanza Amazon RDS all'interno delle sottoreti private. Assicurati che Multi-AZ sia attivato per HA.	Amministratore del cloud
Carica i dati sull'istanza Amazon RDS.	Carica i dati relazionali richiesti dall'applicazione nella tua istanza Amazon RDS. Questo processo varierà in base alle esigenze dell'appl icazione e al modo in cui lo schema del database viene definito e progettato.	Amministratore cloud, DBA

Crea i componenti Amazon ECS

Operazione	Description	Competenze richieste
Crea un cluster ECS.	1. Apri la console Amazon ECS e scegli Clusters. 2. Scegli Crea cluster e configura un cluster ECS in base alle specifiche richieste.	Amministratore cloud
Crea le immagini Docker.	Crea le immagini Docker seguendo le istruzioni nella sezione Risorse correlate.	Amministratore cloud
Crea repository Amazon ECR.	1. Sulla console Amazon ECR, scegli Repositories. 2. Scegli Crea repository e inserisci un nome univoco per il tuo repository. 3. Configura il repository in base alle tue specifiche, inclusa la crittografia AWS KMS, se richiesta.	Amministratore del cloud, ingegnere DevOps
Autentica il tuo client Docker per il repository Amazon ECR.	Per autenticare il tuo client Docker per il repository Amazon ECR, esegui il comando "`aws ecr get-login-password` nella CLI di AWS.	Amministratore cloud
Invia le immagini Docker al repository Amazon ECR.	1. Identifica l'immagine Docker che desideri inviare ed esegui il `docker images` comando nella CLI di AWS.	Amministratore cloud

Operazione	Description	Competenze richieste
	2. Aggiungi tag alla tua immagine con il registro Amazon ECR, il repositor y e la combinazione opzionale del nome del tag di immagine. 3. Invia l'immagine Docker eseguendo il comando. `docker push` 4. Ripeti questi passaggi per tutte le immagini richieste.	

Operazione	Description	Competenze richieste
Crea una definizione di attività Amazon ECS.	Per eseguire i container Docker in Amazon ECS è necessaria una definizione di attività. 1. Torna alla console Amazon ECS, scegli Definizio ni attività, quindi scegli Crea nuova definizione di attività. 2. Nella pagina Seleziona compatibilità, seleziona il tipo di avvio che l'attivit à deve utilizzare e scegli Passaggio successivo. **Important** Per informazioni sull'impostazione della definizione dell'attività, consulta «Creazione di una definizione di attività» nella sezione Risorse correlate. Assicurati di fornire le immagini Docker che hai inviato ad Amazon ECR.	Amministratore cloud

Operazione	Description	Competenze richieste
Crea un servizio Amazon ECS.	Crea un servizio Amazon ECS utilizzando il cluster ECS creato in precedenza. Assicurati di scegliere Amazon EC2 come tipo di avvio e scegli la definizione dell'atti vità creata nel passaggio precedente, nonché il gruppo target dell'Application Load Balancer.	Amministratore cloud

Crea un gruppo Amazon EC2 Auto Scaling

Operazione	Description	Competenze richieste
Creazione di una configura zione di avvio.	Apri la EC2 console Amazon e crea una configurazione di avvio. Assicurati che i dati utente contengano il codice per consentire alle EC2 istanze di unirsi al cluster ECS desiderato. Per un esempio del codice richiesto, consulta la sezione Risorse correlate.	Amministratore cloud
Crea un gruppo Amazon EC2 Auto Scaling.	Torna alla EC2 console Amazon e in Auto Scaling, scegli i gruppi Auto Scaling. Configura un gruppo Amazon EC2 Auto Scaling. Assicurati di scegliere le sottoreti private e di avviare la configurazione creata in precedenza.	Amministratore cloud

Configura AWS PrivateLink

Operazione	Description	Competenze richieste
Configura l' PrivateLink endpoint AWS.	1. Sulla console Amazon VPC, crea un endpoint PrivateLink AWS. 2. Associa questo endpoint al Network Load Balancer, che rende l'applicazione ospitata su Amazon ECS disponibile privatamente per i clienti. Per ulteriori informazioni, consulta la sezione Risorse correlate.	Amministratore cloud

Creare un endpoint VPC

Operazione	Description	Competenze richieste
Crea un endpoint VPC.	Crea un endpoint VPC per l'endpoint AWS che hai creato in PrivateLink precedenza. L'endpoint VPC Fully Qualified Domain Name (FQDN) punterà al nome di dominio completo dell'endpoint AWS. PrivateLink Questo crea un'interfaccia di rete elastica per il servizio endpoint VPC a cui gli endpoint DNS possono accedere.	Amministratore cloud

Creazione della funzione Lambda

Operazione	Description	Competenze richieste
Creazione della funzione Lambda	Sulla console AWS Lambda, crea una funzione Lambda per aggiornare gli indirizzi IP di Application Load Balancer come destinazioni per il Network Load Balancer. Per ulteriori informazioni su questo argomento, consulta il post sul blog Using AWS Lambda to enable static IP address for Application Load Balancers.	Sviluppatore di app

Risorse correlate

Crea i sistemi di bilanciamento del carico:

- Usa un Network Load Balancer per Amazon ECS
- Creare un Network Load Balancer
- Usa un Application Load Balancer per Amazon ECS
- Creare un Application Load Balancer

Crea un file system Amazon EFS:

- Crea un file system Amazon EFS
- Crea obiettivi di montaggio in Amazon EFS

Crea un bucket S3:

- Crea un bucket S3

Crea un segreto di Secrets Manager:

- Crea chiavi in AWS KMS
- Crea un segreto in AWS Secrets Manager

Crea un'istanza Amazon RDS:

- Crea un'istanza database Amazon RDS

Crea i componenti Amazon ECS:

- Crea un cluster Amazon ECS
- Crea un'immagine Docker
- Crea un repository Amazon ECR
- Autentica Docker con il repository Amazon ECR
- Invia un'immagine a un repository Amazon ECR
- Crea una definizione di attività Amazon ECS
- Crea un servizio Amazon ECS

Crea un gruppo Amazon EC2 Auto Scaling:

- Crea una configurazione di lancio
- Creare un gruppo con dimensionamento automatico utilizzando una configurazione di avvio
- Istanze di container Bootstrap con dati utente Amazon EC2

Configura AWS PrivateLink:

- Servizi endpoint VPC (AWS) PrivateLink

Crea un endpoint VPC:

- Endpoint VPC di interfaccia (AWS) PrivateLink

Crea la funzione Lambda:

- Creare una funzione Lambda

Altre risorse:

- Utilizzo di indirizzi IP statici per Application Load Balancer
- Accesso sicuro ai servizi tramite AWS PrivateLink

Accedi alle applicazioni container in modo privato su Amazon ECS utilizzando AWS Fargate, PrivateLink AWS e un Network Load Balancer

Kirankumar Chandrashekar, Amazon Web Services

Riepilogo

Questo modello descrive come ospitare privatamente un'applicazione contenitore Docker sul cloud Amazon Web Services (AWS) utilizzando Amazon Elastic Container Service (Amazon ECS) con un tipo di avvio AWS Fargate, con un Network Load Balancer, e accedere all'applicazione utilizzando AWS. PrivateLink Amazon Relational Database Service (Amazon RDS) ospita il database relazionale per l'applicazione in esecuzione su Amazon ECS con alta disponibilità (HA). Puoi usare Amazon Elastic File System (Amazon EFS) se l'applicazione richiede uno storage persistente.

Questo modello utilizza un tipo di avvio Fargate per il servizio Amazon ECS che esegue le applicazioni Docker, con un Network Load Balancer nel front-end. Può quindi essere associato a un endpoint di cloud privato virtuale (VPC) per l'accesso tramite AWS. PrivateLink Questo servizio di endpoint VPC può quindi essere condiviso con altri utenti utilizzando i rispettivi VPCs endpoint VPC.

Puoi usare Fargate con Amazon ECS per eseguire container senza dover gestire server o cluster di istanze Amazon Elastic Compute Cloud (EC2Amazon). Puoi anche utilizzare un gruppo Amazon EC2 Auto Scaling anziché Fargate. Per ulteriori informazioni, consulta Accesso privato alle applicazioni container su Amazon ECS utilizzando AWS PrivateLink e un Network Load Balancer.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- AWS Command Line Interface (AWS CLI) versione 2, installata e configurata su Linux, macOS o Windows
- Docker, installato e configurato su Linux, macOS o Windows
- Un'applicazione in esecuzione su Docker

Architecture

Stack tecnologico

- Amazon CloudWatch
- Amazon Elastic Container Registry (Amazon ECR)
- Amazon ECS
- Amazon EFS
- Amazon RDS
- Amazon Simple Storage Service (Amazon S3)
- AWS Fargate
- AWS PrivateLink
- AWS Secrets Manager
- Application Load Balancer
- Network Load Balancer
- VPC

Automazione e scalabilità

- Puoi usare AWS CloudFormation per creare questo modello utilizzando Infrastructure as Code.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container AWS gestito che è sicuro, scalabile e affidabile.
- Amazon Elastic Container Service (Amazon ECS) è un servizio di gestione dei container veloce e altamente scalabile che semplifica l'esecuzione, l'arresto e la gestione dei container su un cluster.
- Amazon Elastic File System (Amazon EFS) fornisce un file system NFS elastico semplice, scalabile e completamente gestito da utilizzare con i servizi cloud AWS e le risorse locali.
- AWS Fargateè una tecnologia che puoi usare con Amazon ECS per eseguire container senza dover gestire server o cluster di istanze Amazon EC2 .
- Amazon Relational Database Service (Amazon RDS) è un servizio Web che semplifica la configurazione, il funzionamento e la scalabilità di un database relazionale in. Cloud AWS
- Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è uno storage per Internet. È concepito per rendere più accessibili agli sviluppatori risorse informatiche su grande scala per il Web.

- Con Gestione dei segreti AWS puoi sostituire le credenziali nel codice,incluse le password, con una chiamata API a Secrets Manager in modo da recuperare il segreto a livello di codice.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito.
- Elastic Load Balancing (ELB) distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni, come EC2 istanze, contenitori e indirizzi IP, in più zone di disponibilità.

Altri strumenti

- Docker aiuta gli sviluppatori a imballare, spedire ed eseguire facilmente qualsiasi applicazione come contenitore leggero, portatile e autosufficiente.

Epiche

Crea componenti di rete

Operazione	Description	Competenze richieste
Crea un VPC.	1. Accedi alla Console di gestione AWS e apri la console Amazon VPC. Scegli Crea VPC e scegli VPC e altro ancora. 2. Inserisci un nome per il tuo VPC e scegli un intervallo di blocchi CIDR appropriato. 3. Specificate due zone di disponibilità, due sottoreti pubbliche, quattro sottoreti private. Due sottoreti private sono per le attività di Amazon ECS e due sottoreti private sono per i database Amazon RDS.	Amministratore cloud

Operazione	Description	Competenze richieste
	4. Specificare un gateway NAT per ogni zona di disponibilità. 5. Seleziona Crea VPC.	

Crea i sistemi di bilanciamento del carico

Operazione	Description	Competenze richieste
Crea un Network Load Balancer.	1. Apri la EC2 console Amazon e scegli la regione AWS che contiene il tuo VPC. 2. In Load balancing, scegli Load balancer e scegli Crea load balancer. 3. Scegli Network Load Balancer e scegli Crea. 4. Nella pagina Configure load balancer, configura Network Load Balancer e listener. Importante: assicurati di scegliere lo schema del Network Load Balancer come Interno. 5. Scegli le impostazioni di sicurezza applicabili, configura un gruppo di sicurezza e un gruppo target. Scegli IP come tipo di destinazione nella sezione Configura il routing.	Amministratore cloud

Operazione	Description	Competenze richieste
	Assicurati di non registrare un obiettivo. 6. Dopo aver configurato tutte le impostazioni, scegli Avanti: Revisione, quindi scegli Crea. Per informazioni su questa e altre storie, consulta la sezione Risorse correlate.	

Operazione	Description	Competenze richieste
Crea un Application Load Balancer.	1. Sulla EC2 console Amazon, scegli la stessa regione che contiene il tuo VPC. 2. In Load balancing, scegli Load balancer e scegli Crea load balancer. 3. Scegliete Application Load Balancer e scegliete Create. 4. **Important** Configura l'Application Load Balancer e il relativo listener. Assicurati di scegliere lo schema di Application Load Balancer come Interno. 5. Scegliete le impostazioni di sicurezza applicabili, configurate un gruppo di sicurezza e un gruppo target. Scegli IP come tipo di destinazione nella sezione Configura il routing. Assicurati di non registrare un obiettivo. 6. Dopo aver configurato tutte le impostazioni, scegli Avanti: Revisione, quindi scegli Crea.	Amministratore cloud

Creare un file system Amazon EFS

Operazione	Description	Competenze richieste
Crea un file system Amazon EFS.	1. Apri la console Amazon EFS e scegli Crea file system. 2. Nella finestra di dialogo Crea file system, inserisci un nome per il tuo file system e scegli il tuo VPC. 3. Scegli Crea per creare il file system. 4. Configura e configura il tuo file system Amazon EFS.	Amministratore cloud
Monta gli obiettivi per le sottoreti.	1. Torna alla console Amazon EFS e scegli File system. La pagina File system mostra i file system Amazon EFS presenti nel tuo account. 2. Scegli il file system che hai creato e scegli Gestisci per visualizzare la zona di disponibilità. 3. Per aggiungere una destinazione di montaggio , scegli Aggiungi destinazi one di montaggio e aggiungi le quattro sottoreti private che hai creato.	Amministratore cloud
Verifica che le sottoreti siano montate come destinazioni.	1. Sulla console Amazon EFS, scegli File system.	Amministratore cloud

Operazione	Description	Competenze richieste
	2. Scegli Rete per visualizzare l'elenco dei target di montaggio esistenti. Assicurati che questi includano le quattro sottoreti che hai creato.	

Creare un bucket S3

Operazione	Description	Competenze richieste
Crea un bucket S3.	Apri la console Amazon S3 e crea un bucket S3 per archiviare gli asset statici dell'applicazione, se necessario.	Amministratore cloud

Crea un segreto di Secrets Manager

Operazione	Description	Competenze richieste
Crea una chiave AWS KMS per crittografare il segreto di Secrets Manager.	Apri la console AWS Key Management Service (AWS KMS) e crea una chiave KMS.	Amministratore del cloud
Crea un segreto di Secrets Manager per archiviare la password di Amazon RDS.	1. Apri la console AWS Secrets Manager e crea un nuovo segreto scegliendo Archivia un nuovo segreto. 2. Scegli la chiave KMS che hai creato e archivia il tuo nuovo segreto.	Amministratore cloud

Crea un'istanza Amazon RDS

Operazione	Description	Competenze richieste
Creazione di un gruppo di sottoreti del database.	1. Apri la console Amazon RDS e scegli Gruppi di sottorete. 2. Scegli Crea gruppo di sottoreti DB e inserisci un nome e una descrizione per il tuo gruppo di sottoreti DB. 3. Scegli il VPC che hai creato in precedenza e scegli le zone di disponibilità e le sottoreti. Quindi, scegli Crea.	Amministratore cloud
Crea un'istanza Amazon RDS.	Crea e configura un'istanza Amazon RDS all'interno delle sottoreti private. Assicurati che Multi-AZ sia attivato per l'alta disponibilità (HA).	Amministratore cloud
Carica i dati sull'istanza Amazon RDS.	Carica i dati relazionali richiesti dall'applicazione nella tua istanza Amazon RDS. Questo processo varierà in base alle esigenze dell'appl icazione e al modo in cui lo schema del database viene definito e progettato.	DBA

Crea i componenti Amazon ECS

Operazione	Description	Competenze richieste
Crea un cluster ECS.	1. Apri la console Amazon ECS e scegli Clusters. 2. Scegli Crea cluster e configura un cluster ECS in base alle specifiche richieste.	Amministratore cloud
Crea le immagini Docker.	Crea le immagini Docker seguendo le istruzioni nella AWS documentazione.	Amministratore cloud
Crea un repository Amazon ECR.	1. Apri la console Amazon ECR e seleziona Repositories. 2. Scegli Crea repository e inserisci un nome univoco per il tuo repository. 3. Configura il repository in base alle tue specifiche, inclusa la crittografia AWS KMS, se richiesta.	Amministratore del cloud, ingegnere DevOps
Invia le immagini Docker al repository Amazon ECR.	1. Identifica l'immagine Docker che desideri inviare ed esegui il `docker images` comando nella CLI di AWS. 2. Aggiungi tag alla tua immagine con il registro Amazon ECR, il repository e la combinazione opzionale del nome del tag di immagine.	Amministratore cloud

Operazione	Description	Competenze richieste
	3. Invia l'immagine Docker eseguendo il comando. `docker push` 4. Ripeti questi passaggi per tutte le immagini richieste.	

Operazione	Description	Competenze richieste
Crea una definizione di attività Amazon ECS.	Per eseguire i container Docker in Amazon ECS è necessaria una definizione di attività. 1. Torna alla console Amazon ECS, scegli Definizio ni attività, quindi scegli Crea nuova definizione di attività. 2. Nella pagina Seleziona compatibilità, seleziona il tipo di avvio che l'attivit à deve utilizzare e scegli Passaggio successivo. **Important** Per informazioni sull'impostazione della definizione dell'attività, consulta «Creazione di una definizione di attività» nella sezione Risorse correlate. Assicurati di fornire le immagini Docker che hai inviato ad Amazon ECR.	Amministratore cloud

Operazione	Description	Competenze richieste
Crea un servizio ECS e scegli Fargate come tipo di lancio.	1. Crea un servizio Amazon ECS utilizzando il cluster ECS creato in precedenza. Assicurati di scegliere Fargate come tipo di lancio. 2. Scegliete la definizione dell'attività creata nel passaggio precedente e scegliete il gruppo target dell'Application Load Balancer.	Amministratore cloud

Configura AWS PrivateLink

Operazione	Description	Competenze richieste
Configura l' PrivateLink endpoint AWS.	1. Apri la console Amazon VPC e crea un AWS PrivateLink endpoint. 2. Associa questo endpoint al Network Load Balancer, che rende l'applicazione ospitata su Amazon ECS disponibile privatamente per i clienti.	Amministratore cloud

Creare un endpoint VPC

Operazione	Description	Competenze richieste
Crea un endpoint VPC.	Crea un endpoint VPC per l'endpoint AWS che hai creato in PrivateLink precedenz	Amministratore cloud

Operazione	Description	Competenze richieste
	a. L'endpoint VPC Fully Qualified Domain Name (FQDN) punterà al nome di dominio completo dell'endpoint AWS. PrivateLink Questo crea un'interfaccia di rete elastica per il servizio endpoint VPC a cui possono accedere gli endpoint del Domain Name Service.	

Imposta l'obiettivo

Operazione	Description	Competenze richieste
Aggiungete l'Application Load Balancer come destinazione.	Per aggiungere l'Application Load Balancer come destinazione per il Network Load Balancer, segui le istruzioni nella documentazione.AWS	Sviluppatore di app

Risorse correlate

Crea i load balancer:

- Usa un Network Load Balancer per Amazon ECS
- Creare un Network Load Balancer
- Usa un Application Load Balancer per Amazon ECS
- Creare un Application Load Balancer

Crea un file system Amazon EFS:

- Crea un file system Amazon EFS

- Crea obiettivi di montaggio in Amazon EFS

Crea un segreto di Secrets Manager:

- Crea chiavi in AWS KMS
- Crea un segreto in AWS Secrets Manager

Crea un'istanza Amazon RDS:

- Crea un'istanza database Amazon RDS

Crea i componenti Amazon ECS

- Crea un repository Amazon ECR
- Autentica Docker con il repository Amazon ECR
- Invia un'immagine a un repository Amazon ECR
- Crea una definizione di attività Amazon ECS
- Crea un servizio Amazon ECS

Altre risorse:

- Accesso sicuro ai servizi tramite AWS PrivateLink

Accedi alle applicazioni container in modo privato su Amazon EKS utilizzando AWS PrivateLink e un Network Load Balancer

Kirankumar Chandrashekar, Amazon Web Services

Riepilogo

Questo modello descrive come ospitare privatamente un'applicazione container Docker su Amazon Elastic Kubernetes Service (Amazon EKS) con un Network Load Balancer e accedere all'applicazione utilizzando AWS. PrivateLink Puoi quindi utilizzare una rete privata per accedere in modo sicuro ai servizi sul cloud Amazon Web Services (AWS).

Il cluster Amazon EKS che esegue le applicazioni Docker, con un Network Load Balancer sul front-end, può essere associato a un endpoint di cloud privato virtuale (VPC) per l'accesso tramite AWS. PrivateLink Questo servizio di endpoint VPC può quindi essere condiviso con altri utenti utilizzando i rispettivi VPCs endpoint VPC.

La configurazione descritta da questo modello è un modo sicuro per condividere l'accesso alle applicazioni VPCs tra account AWS. Non richiede configurazioni di connettività o routing speciali, poiché la connessione tra gli account consumer e provider si trova sulla spina dorsale globale di AWS e non attraversa la rete Internet pubblica.

Prerequisiti e limitazioni

Prerequisiti

- Docker, installato e configurato su Linux, macOS o Windows.
- Un'applicazione in esecuzione su Docker.
- Un account AWS attivo.
- AWS Command Line Interface (AWS CLI) versione 2, installata e configurata su Linux, macOS o Windows.
- Un cluster Amazon EKS esistente con sottoreti private etichettate e configurato per ospitare applicazioni. Per ulteriori informazioni, consulta Subnet tagging nella documentazione di Amazon EKS.
- Kubectl, installato e configurato per accedere alle risorse sul tuo cluster Amazon EKS. Per ulteriori informazioni, consulta Installazione di kubectl nella documentazione di Amazon EKS.

Architecture

Stack tecnologico

- Amazon EKS
- AWS PrivateLink
- Network Load Balancer

Automazione e scalabilità

- I manifesti di Kubernetes possono essere tracciati e gestiti su un repository basato su Git e distribuiti utilizzando l'integrazione continua e la distribuzione continua (CI/CD) in AWS. CodePipeline
- Puoi usare AWS CloudFormation per creare questo modello utilizzando Infrastructure as Code (IaC).

Tools (Strumenti)

- AWS CLI — AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire con i servizi AWS utilizzando i comandi nella shell della riga di comando.
- Elastic Load Balancing: Elastic Load Balancing distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni, come istanze Amazon Elastic Compute Cloud (EC2Amazon), contenitori e indirizzi IP, in una o più zone di disponibilità.

- Amazon EKS — Amazon Elastic Kubernetes Service (Amazon EKS) è un servizio gestito che puoi usare per eseguire Kubernetes su AWS senza dover installare, gestire e mantenere il tuo piano di controllo o i tuoi nodi Kubernetes.
- Amazon VPC: Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito.
- Kubectl — Kubectl è un'utilità da riga di comando per eseguire comandi su cluster Kubernetes.

Epiche

Distribuisci i file manifest di distribuzione e servizio di Kubernetes

Operazione	Description	Competenze richieste

Crea il file manifesto di distribuzione di Kubernetes.	Crea un file manifesto di distribuzione modificando il seguente file di esempio in base alle tue esigenze.
	
	`apiVersion: apps/v1`
	`kind: Deployment`
	`metadata:`
	`  name: sample-app`
	`spec:`
	`  replicas: 3`
	`  selector:`
	`  matchLabels:`
	`  app: nginx`
	`  template:`
	`  metadata:`
	`  labels:`
	`  app: nginx`
	`  spec:`
	`  containers:`
	`  - name: nginx`
	`  image:`
	`public.ecr.aws/z9d2n7e1/nginx:1.19.5`
	`  ports:`
	`  - name: http`
	`  containerPort: 80`
	
	Note
	Questo è un file di configurazione di esempio di NGINX che viene distribuito
		DevOps ingegnere

Operazione	Description	Competenze richieste
	utilizzando l'immagine NGINX Docker. Per ulteriori informazioni, consulta Come usare l'immagine ufficiale di NGINX Docker nella documentazione Docker.	
Distribuisci il file manifesto di distribuzione di Kubernetes.	Esegui il comando seguente per applicare il file manifest di distribuzione al tuo cluster Amazon EKS: `kubectl apply -f <your_deployment_file_name>`	DevOps ingegnere

Operazione	Description	Competenze richieste

Crea il file manifesto del servizio Kubernetes.	Crea un file di manifesto del servizio modificando il seguente file di esempio in base alle tue esigenze.
	
	`apiVersion: v1`
	`kind: Service`
	`metadata:`
	`  name: sample-service`
	`  annotations:`
	`  service.beta.kuber`
	`netes.io/aws-load-`
	`balancer-type: nlb`
	`  service.beta.kuber`
	`netes.io/aws-load-`
	`balancer-internal:`
	` "true"`
	`spec:`
	`  ports:`
	`  - port: 80`
	`  targetPort: 80`
	`  protocol: TCP`
	`  type: LoadBalancer`
	`  selector:`
	`  app: nginx`
	
	⚠ Important
	Assicurati di aver incluso quanto segue `annotations` per definire un Network Load Balancer interno:
	
	`service.beta.kuber`
	`netes.io/aws-load-`
	`balancer-type: nlb`
		DevOps ingegnere

Operazione	Description	Competenze richieste
	`service.beta.kubernetes.io/aws-load-balancer-internal: "true"`	
Distribuisci il file di manifesto del servizio Kubernetes.	Esegui il comando seguente per applicare il file manifest del servizio al tuo cluster Amazon EKS: `kubectl apply -f <your_service_file_name>`	DevOps ingegnere

Crea gli endpoint

Operazione	Description	Competenze richieste
Registra il nome del Network Load Balancer.	Esegui il comando seguente per recuperare il nome del Network Load Balancer: `kubectl get svc sample-service -o wide` Registra il nome del Network Load Balancer, necessario per creare un PrivateLink endpoint AWS.	DevOps ingegnere
Crea un PrivateLink endpoint AWS.	Accedi alla Console di gestione AWS, apri la console Amazon VPC e crea un endpoint PrivateLink AWS. Associando questo endpoint	Amministratore cloud

Operazione	Description	Competenze richieste
	al Network Load Balancer, l'applicazione sarà disponibile privatamente per i clienti. Per ulteriori informazioni, consulta VPC endpoint services (AWS PrivateLink) nella documentazione di Amazon VPC. **Important** Se l'account consumer richiede l'accesso all'applicazione, l'ID dell'account AWS dell'account consumer deve essere aggiunto all'elenco dei principali consentiti per la configurazione degli PrivateLink endpoint AWS. Per ulteriori informazioni, consulta Aggiungere e rimuovere le autorizzazioni per il servizio endpoint nella documentazione di Amazon VPC.	

Operazione	Description	Competenze richieste
Crea un endpoint VPC.	Sulla console Amazon VPC, scegli Endpoint Services, quindi scegli Create Endpoint Service. Crea un endpoint VPC per l'endpoint AWS. PrivateLink Il nome di dominio completo (FQDN) dell'endpoint VPC punta al nome di dominio completo per l'endpoint AWS. PrivateLink Questo crea un'interfaccia di rete elastica per il servizio endpoint VPC a cui gli endpoint DNS possono accedere.	Amministratore cloud

Risorse correlate

- Utilizzando l'immagine ufficiale di NGINX Docker
- Bilanciamento del carico di rete su Amazon EKS
- Creazione di servizi endpoint VPC (AWS) PrivateLink
- Aggiungere e rimuovere le autorizzazioni per il servizio endpoint

Automatizza i backup per le istanze DB di Amazon RDS for PostgreSQL utilizzando AWS Batch

Kirankumar Chandrashekar, Amazon Web Services

Riepilogo

Il backup dei database PostgreSQL è un'attività importante e in genere può essere completata con l'utilità pg_dump, che utilizza il comando COPY per impostazione predefinita per creare uno schema e un dump dei dati di un database PostgreSQL. Tuttavia, questo processo può diventare ripetitivo se sono necessari backup regolari per più database PostgreSQL. Se i tuoi database PostgreSQL sono ospitati nel cloud, puoi anche sfruttare la funzionalità di backup automatico fornita da Amazon Relational Database Service (Amazon RDS) anche per PostgreSQL. Questo modello descrive come automatizzare i backup regolari per le istanze DB Amazon RDS for PostgreSQL utilizzando l'utilità pg_dump.

Nota: le istruzioni presuppongono che tu stia utilizzando Amazon RDS. Tuttavia, puoi utilizzare questo approccio anche per i database PostgreSQL ospitati all'esterno di Amazon RDS. Per eseguire i backup, la funzione AWS Lambda deve essere in grado di accedere ai database.

Un evento Amazon CloudWatch Events basato sul tempo avvia una funzione Lambda che cerca tag di backup specifici applicati ai metadati delle istanze DB PostgreSQL su Amazon RDS. Se le istanze DB PostgreSQL hanno il tag DBDump BKP:Automated = Active e altri tag di backup richiesti, la funzione Lambda invia singoli job per ogni backup del database ad AWS Batch.

AWS Batch elabora questi processi e carica i dati di backup in un bucket Amazon Simple Storage Service (Amazon S3). Questo modello utilizza un Dockerfile e un file entrypoint.sh per creare un'immagine del contenitore Docker che viene utilizzata per eseguire backup nel job AWS Batch. Una volta completato il processo di backup, AWS Batch registra i dettagli del backup in una tabella di inventario su Amazon DynamoDB. Come ulteriore protezione, un evento CloudWatch Events avvia una notifica Amazon Simple Notification Service (Amazon SNS) se un processo fallisce in AWS Batch.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.

- Un ambiente di elaborazione gestito o non gestito esistente. Per ulteriori informazioni, consulta Ambienti di calcolo gestiti e non gestiti nella documentazione di AWS Batch.
- Immagine Docker AWS Command Line Interface (CLI) versione 2, installata e configurata.
- Istanze database Amazon RDS for PostgreSQL esistenti.
- Un bucket S3 esistente.
- Docker, installato e configurato su Linux, macOS o Windows.
- Familiarità con la programmazione in Lambda.

Architecture

Stack tecnologico

- CloudWatch Eventi Amazon
- Amazon DynamoDB
- Amazon Elastic Container Registry (Amazon ECR)
- Amazon RDS
- Amazon SNS
- Simple Storage Service (Amazon S3)
- AWS Batch
- AWS Key Management Service (AWS KMS)
- AWS Lambda
- AWS Secrets Manager
- Docker

Tools (Strumenti)

- Amazon CloudWatch Events — CloudWatch Events offre un flusso quasi in tempo reale di eventi di sistema che descrivono i cambiamenti nelle risorse AWS.
- Amazon DynamoDB — DynamoDB è un servizio di database NoSQL completamente gestito che offre prestazioni veloci e prevedibili con una scalabilità perfetta.
- Amazon ECR — Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container AWS gestito che è sicuro, scalabile e affidabile.

- Amazon RDS — Amazon Relational Database Service (Amazon RDS) è un servizio Web che semplifica la configurazione, il funzionamento e la scalabilità di un database relazionale nel cloud AWS.
- Amazon SNS — Amazon Simple Notification Service (Amazon SNS) è un servizio gestito che fornisce il recapito dei messaggi dagli editori agli abbonati.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è uno storage per Internet.
- AWS Batch: AWS Batch ti aiuta a eseguire carichi di lavoro di elaborazione in batch sul cloud AWS.
- AWS KMS — AWS Key Management Service (AWS KMS) è un servizio gestito che semplifica la creazione e il controllo delle chiavi di crittografia utilizzate per crittografare i dati.
- AWS Lambda — Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza effettuare il provisioning o gestire server.
- AWS Secrets Manager — Secrets Manager ti aiuta a sostituire le credenziali hardcoded nel codice, incluse le password, con una chiamata API a Secrets Manager per recuperare il segreto a livello di codice.
- Docker — Docker aiuta gli sviluppatori a imballare, spedire ed eseguire facilmente qualsiasi applicazione come contenitore leggero, portatile e autosufficiente.

Le tue istanze DB PostgreSQL su Amazon RDS devono avere tag applicati ai relativi metadati.
La funzione Lambda cerca i tag per identificare le istanze DB di cui eseguire il backup e in genere vengono utilizzati i tag seguenti.

Tag	Description
BKP: automatizzato = attivo DBDump	Identifica un'istanza DB Amazon RDS come candidata per i backup.
bkp: = AutomatedBackupSecret <secret_name>	Identifica il segreto di Secrets Manager che contiene le credenziali di accesso di Amazon RDS.
BKP: bucket S3 automatizzato = DBDump <s3_bucket_name>	Identifica il bucket S3 a cui inviare i backup.
BKP: frequenza automatizzata DBDump	Identifica la frequenza e gli orari in cui eseguire il backup dei database.

BKP: Tempo automatizzato DBDump

bkp: comando pgdump = <pgdump_command>	Identifica i database per i quali devono essere eseguiti i backup.

Epiche

Creare una tabella di inventario in DynamoDB

Operazione	Description	Competenze richieste
Crea una tabella in DynamoDB.	Accedi alla Console di gestione AWS, apri la console Amazon DynamoDB e crea una tabella. Per assistenza su questa e altre storie, consulta la sezione Risorse correlate.	Amministratore del cloud, amministratore del database
Conferma che la tabella è stata creata.	Esegui il comando `aws dynamodb describe-table --table-name <table-name> | grep TableStatus` . Se la tabella esiste, il comando restituir à il `"TableStatus": "ACTIVE",` risultato.	Amministratore del cloud, amministratore del database

Crea un argomento SNS per gli eventi di job non riusciti in AWS Batch

Operazione	Description	Competenze richieste
Creare un argomento SNS.	Apri la console Amazon SNS, scegli Argomenti e crea un argomento SNS con il nome. `JobFailedAlert` Sottoscri vi un indirizzo e-mail attivo	Amministratore del cloud

Operazione	Description	Competenze richieste
	all'argomento e controlla la tua casella di posta elettronica per confermare l'e-mail di iscrizion e a SNS da AWS Notifications.	
Crea una regola relativa agli eventi di lavoro non riusciti per AWS Batch.	Apri la CloudWatch console Amazon, scegli Eventi, quindi scegli Crea regola. Scegli Mostra opzioni avanzate e scegli Modifica. Per Crea un pattern che seleziona gli eventi da elaborare in base ai tuoi obiettivi, sostituisci il testo esistente con il codice «Failed job event» nella sezione Informazioni aggiuntiv e. Questo codice definisce una regola CloudWatch Events che viene avviata quando AWS Batch ha un `Failed` evento.	Amministratore cloud
Aggiungi l'obiettivo della regola dell'evento.	In Target, scegli Aggiungi obiettivi e scegli l'argomen to `JobFailedAlert` SNS. Configura i dettagli rimanenti e crea la regola Cloudwatch Events.	Amministratore cloud

Crea un'immagine Docker e inviala a un repository Amazon ECR

Operazione	Description	Competenze richieste
Crea un repository Amazon ECR.	Apri la console Amazon ECR e scegli la regione AWS in cui	Amministratore cloud

Operazione	Description	Competenze richieste
	desideri creare il tuo repository. Scegli Repositories, quindi scegli Crea repository. Configura il repository in base alle tue esigenze.	
Scrivere un Dockerfile.	Accedi a Docker e usa «Sample Dockerfile» e «Sample entrypoint.sh file» dalla sezione Informazioni aggiuntive per creare un Dockerfile.	DevOps ingegnere
Crea un'immagine Docker e inviala al repository Amazon ECR.	Crea il Dockerfile in un'immagine Docker e invialo al repository Amazon ECR. Per informazioni su questa storia, consulta la sezione Risorse correlate.	DevOps ingegnere

Crea i componenti AWS Batch

Operazione	Description	Competenze richieste
Crea una definizione di processo AWS Batch.	Apri la console AWS Batch e crea una definizione di processo che includa l'URI (Uniform Resource Identifier) del repository Amazon ECR come proprietà. `Image`	Amministratore cloud
Configura la coda dei job di AWS Batch.	Nella console AWS Batch, scegli Job queues, quindi scegli Create queue. Crea una coda di lavoro che memorizze	Amministratore cloud

Operazione	Description	Competenze richieste
	rà i lavori fino a quando AWS Batch non li eseguirà sulle risorse all'interno del tuo ambiente di calcolo. Importante: assicurati di scrivere la logica per AWS Batch per registrare i dettagli del backup nella tabella di inventario di DynamoDB.	

Creare e pianificare una funzione Lambda

Operazione	Description	Competenze richieste
Crea una funzione Lambda per cercare i tag.	Crea una funzione Lambda che cerca i tag nelle tue istanze DB PostgreSQL e identifica i candidati al backup. Assicurati che la tua funzione Lambda sia in grado di identificare il `bkp:AutomatedDBDump = Active` tag e tutti gli altri tag richiesti . Importante: la funzione Lambda deve anche essere in grado di aggiungere lavori alla coda di lavori di AWS Batch.	DevOps ingegnere
Crea un evento CloudWatch Events basato sul tempo.	Apri la CloudWatch console Amazon e crea un evento CloudWatch Events che utilizzi un'espressione cron per eseguire la funzione Lambda a intervalli regolari. Important	Amministratore cloud

Operazione	Description	Competenze richieste
	e: tutti gli eventi pianificati utilizzano il fuso orario UTC.	

Prova l'automazione del backup

Operazione	Description	Competenze richieste
Crea una chiave Amazon KMS.	Apri la console Amazon KMS e crea una chiave KMS che può essere utilizzata per crittografare le credenziali Amazon RDS archiviate in AWS Secrets Manager.	Amministratore cloud
Crea un segreto di AWS Secrets Manager.	Apri la console AWS Secrets Manager e archivia le credenziali del database Amazon RDS for PostgreSQL come segreto.	Amministratore cloud
Aggiungi i tag richiesti alle istanze DB PostgreSQL.	**Important** Apri la console Amazon RDS e aggiungi tag alle istanze DB PostgreSQL di cui desideri eseguire il backup automatico. Puoi utilizzare i tag della tabella nella sezione Strumenti. Se hai bisogno di backup da più database PostgreSQL all'inter	Amministratore cloud

Operazione	Description	Competenze richieste
	no della stessa istanza Amazon RDS, usali come valore per `-d test:-d test1` il tag. `bkp:pgdum pcommand teste` sono nomi di database. `test1` Assicurati che non ci sia spazio dopo i due punti (:).	
Verifica l'automazione del backup.	Per verificare l'automazione del backup, puoi richiamare la funzione Lambda o attendere l'inizio della pianificazione del backup. Una volta completato il processo di backup, verifica che la tabella di inventario di DynamoDB contenga una voce di backup valida per le tue istanze DB PostgreSQL. Se corrispondono, il processo di automazione del backup ha esito positivo.	Amministratore cloud

Risorse correlate

Creare una tabella di inventario in DynamoDB

- Creare una tabella Amazon DynamoDB

Crea un argomento SNS per gli eventi di job non riusciti in AWS Batch

- Crea un argomento Amazon SNS
- Invia avvisi SNS per eventi di lavoro non riusciti in AWS Batch

Crea un'immagine Docker e inviala a un repository Amazon ECR

- Crea un repository Amazon ECR
- Scrivi un Dockerfile, crea un'immagine Docker e inviala ad Amazon ECR

Crea i componenti AWS Batch

- Crea una definizione di processo AWS Batch
- Configura il tuo ambiente di calcolo e la coda di lavoro di AWS Batch
- Crea una coda di lavoro in AWS Batch

Creare una funzione Lambda

- Crea una funzione Lambda e scrivi codice
- Usare Lambda con DynamoDB

Crea un evento Events CloudWatch

- Crea un CloudWatch evento Events basato sul tempo
- Usa le espressioni cron in Cloudwatch Events

Prova l'automazione del backup

- Crea una chiave Amazon KMS
- Crea un segreto di Secrets Manager
- Aggiungere tag a un'istanza Amazon RDS

Informazioni aggiuntive

Evento di processo non riuscito:

```
{
  "detail-type": [
    "Batch Job State Change"
  ],
  "source": [
    "aws.batch"
  ],
  "detail": {
    "status": [
      "FAILED"
    ]
  }
}
```

Dockerfile di esempio:

```
FROM alpine:latest
RUN apk --update add py-pip postgresql-client jq bash && \
pip install awscli && \
rm -rf /var/cache/apk/*
ADD entrypoint.sh /usr/bin/
RUN chmod +x /usr/bin/entrypoint.sh
ENTRYPOINT ["entrypoint.sh"]
```

File entrypoint.sh di esempio:

```
 #!/bin/bash
set -e
DATETIME=`date +"%Y-%m-%d_%H_%M"`
FILENAME=RDS_PostGres_dump_${RDS_INSTANCE_NAME}
FILE=${FILENAME}_${DATETIME}
```

```
aws configure --profile new-profile set role_arn arn:aws:iam::${TargetAccountId}:role/
${TargetAccountRoleName}
aws configure --profile new-profile set credential_source EcsContainer

echo "Central Account access provider IAM role is: "
aws sts get-caller-identity

echo "Target Customer Account access provider IAM role is: "
aws sts get-caller-identity --profile new-profile

securestring=$(aws secretsmanager get-secret-value --secret-id $SECRETID --output json
 --query 'SecretString' --region=$REGION --profile new-profile)

if [[ ${securestring} ]]; then
    echo "successfully accessed secrets manager and got the credentials"
    export PGPASSWORD=$(echo $securestring | jq --raw-output | jq -r '.DB_PASSWORD')
    PGSQL_USER=$(echo $securestring | jq --raw-output | jq -r '.DB_USERNAME')
    echo "Executing pg_dump for the PostGres endpoint ${PGSQL_HOST}"
    # pg_dump -h $PGSQL_HOST -U $PGSQL_USER -n dms_sample | gzip -9 -c  | aws s3 cp -
 --region=$REGION  --profile new-profile s3://$BUCKET/$FILE
    # in="-n public:-n private"
    IFS=':' list=($EXECUTE_COMMAND);
    for command in "${list[@]}";
      do
        echo $command;
        pg_dump -h $PGSQL_HOST -U $PGSQL_USER ${command} | gzip -9 -c  | aws s3 cp - --
region=$REGION --profile new-profile s3://${BUCKET}/${FILE}-${command}".sql.gz"
        echo $?;
        if  [[ $? -ne 0 ]]; then
            echo "Error occurred in database backup process. Exiting now....."
            exit 1
        else
            echo "Postgresql dump was successfully taken for the RDS endpoint
 ${PGSQL_HOST} and is uploaded to the following S3 location s3://${BUCKET}/${FILE}-
${command}.sql.gz"
            #write the details into the inventory table in central account
            echo "Writing to DynamoDB inventory table"
            aws dynamodb put-item --table-name ${RDS_POSTGRES_DUMP_INVENTORY_TABLE} --
region=$REGION --item '{ "accountId": { "S": "'"${TargetAccountId}"'" }, "dumpFileUrl":
 {"S": "'"s3://${BUCKET}/${FILE}-${command}.sql.gz"'" }, "DumpAvailableTime": {"S":
 "'"`date +"%Y-%m-%d::%H::%M::%S"` UTC"'"}}'
            echo $?
            if  [[ $? -ne 0 ]]; then
```

```
                echo "Error occurred while putting item to DynamoDb Inventory Table.
 Exiting now....."
                exit 1
            else
                echo "Successfully written to DynamoDb Inventory Table
 ${RDS_POSTGRES_DUMP_INVENTORY_TABLE}"
            fi
        fi
      done;
else
    echo "Something went wrong {$?}"
    exit 1
fi

exec "$@"
```

Automatizza la distribuzione di Node Termination Handler in Amazon EKS utilizzando una pipeline CI/CD

Sandip Gangapadhyay, Sandeep Gawande, Viyoma Sachdeva, Pragtideep Singh e John Vargas, Amazon Web Services

Riepilogo

Avviso: AWS non CodeCommit è più disponibile per i nuovi clienti. I clienti esistenti di AWS CodeCommit possono continuare a utilizzare il servizio normalmente. Ulteriori informazioni

Sul cloud Amazon Web Services (AWS), puoi utilizzare AWS Node Termination Handler, un progetto open source, per gestire senza problemi l'arresto delle istanze Amazon Elastic Compute Cloud (Amazon EC2) all'interno di Kubernetes. AWS Node Termination Handler aiuta a garantire che il piano di controllo di Kubernetes risponda in modo appropriato agli eventi che possono rendere l'istanza non disponibile. EC2 Tali eventi includono quanto segue:

- EC2 manutenzione programmata dell'istanza
- Interruzioni delle istanze Amazon EC2 Spot
- Auto Scaling, ridimensionamento del gruppo in
- Ribilanciamento del gruppo Auto Scaling tra le zone di disponibilità
- EC2 terminazione dell'istanza tramite l'API o la Console di gestione AWS

Se un evento non viene gestito, il codice dell'applicazione potrebbe non interrompersi correttamente. Inoltre, potrebbe essere necessario più tempo per ripristinare la piena disponibilità o programmare accidentalmente il lavoro sui nodi che non funzionano. `aws-node-termination-handler`(NTH) può funzionare in due diverse modalità: Instance Metadata Service (IMDS) o Queue Processor. Per ulteriori informazioni sulle due modalità, consultate il file Readme.

Questo modello utilizza AWS CodeCommit e automatizza la distribuzione di NTH utilizzando Queue Processor attraverso una pipeline di integrazione e distribuzione continue (CI/CD).

> **Note**
>
> Se utilizzi gruppi di nodi gestiti da EKS, non hai bisogno di. `aws-node-termination-handler`

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un browser Web supportato per l'uso con la Console di gestione AWS. Consulta l'elenco dei browser supportati.
- AWS Cloud Development Kit (AWS CDK) installato.
- `kubectl`, lo strumento da riga di comando Kubernetes, installato.
- `eksctl`, l'AWS Command Line Interface (AWS CLI) per Amazon Elastic Kubernetes Service (Amazon EKS), installata.
- Un cluster EKS in esecuzione con versione 1.20 o successiva.
- Un gruppo di nodi autogestito collegato al cluster EKS. Per creare un cluster Amazon EKS con un gruppo di nodi autogestito, esegui il comando seguente.

```
eksctl create cluster --managed=false --region <region> --name <cluster_name>
```

  Per ulteriori informazioni su`eksctl`, consulta la documentazione di eksctl.
- Provider AWS Identity and Access Management (IAM) OpenID Connect (OIDC) per il tuo cluster. Per ulteriori informazioni, consulta Creazione di un provider IAM OIDC per il cluster.

Limitazioni

- È necessario utilizzare una regione AWS che supporti il servizio Amazon EKS.

Versioni del prodotto

- Kubernetes versione 1.20 o successiva
- `eksctl`versione 0.107.0 o successiva
- AWS CDK versione 2.27.0 o successiva

Architecture

Stack tecnologico Target

- Un cloud privato virtuale (VPC)

- Un cluster EKS
- Amazon Simple Queue Service (Amazon SQS)
- IAM
- Kubernetes

Architettura Target

Il diagramma seguente mostra la visualizzazione di alto livello dei end-to-end passaggi quando viene avviata la terminazione del nodo.

Il flusso di lavoro mostrato nel diagramma è costituito dai seguenti passaggi di alto livello:

1. L'evento di terminazione dell' EC2 istanza di ridimensionamento automatico viene inviato alla coda SQS.
2. L'NTH Pod monitora la presenza di nuovi messaggi nella coda SQS.
3. L'NTH Pod riceve il nuovo messaggio ed esegue le seguenti operazioni:
   - Collega il nodo in modo che il nuovo pod non venga eseguito sul nodo.
   - Drena il nodo, in modo che il pod esistente venga evacuato
   - Invia un segnale hook del ciclo di vita al gruppo Auto Scaling in modo che il nodo possa essere terminato.

Automazione e scalabilità

- Il codice è gestito e distribuito da AWS CDK, supportato da AWS CloudFormation nested stacks.
- Il piano di controllo di Amazon EKS funziona su più zone di disponibilità per garantire un'elevata disponibilità.
- Per la scalabilità automatica, Amazon EKS supporta Kubernetes Cluster Autoscaler e Karpenter.

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software che aiuta a definire e fornire l'infrastruttura cloud AWS in codice.

- AWS CodeBuild è un servizio di build completamente gestito che ti aiuta a compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS CodeCommit è un servizio di controllo delle versioni che consente di archiviare e gestire archivi Git in modo privato, senza dover gestire il proprio sistema di controllo del codice sorgente.
- AWS ti CodePipeline aiuta a modellare e configurare rapidamente le diverse fasi di un rilascio di software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire Kubernetes su AWS senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- Amazon EC2 Auto Scaling ti aiuta a mantenere la disponibilità delle applicazioni e ti consente di aggiungere o rimuovere automaticamente EC2 istanze Amazon in base alle condizioni da te definite.
- Amazon Simple Queue Service (Amazon SQS) fornisce una coda ospitata sicura, durevole e disponibile che ti aiuta a integrare e disaccoppiare sistemi e componenti software distribuiti.

Altri strumenti

- kubectl è uno strumento da riga di comando di Kubernetes per eseguire comandi su cluster Kubernetes. Puoi usare kubectl per distribuire applicazioni, ispezionare e gestire le risorse del cluster e visualizzare i log.

Codice

Il codice per questo pattern è disponibile nel repository su .com. deploy-nth-to-eks GitHub Il repository di codice contiene i seguenti file e cartelle.

- `nth folder`— Il grafico Helm, i file di valori e gli script per scansionare e distribuire il CloudFormation modello AWS per Node Termination Handler.
- `config/config.json`— Il file dei parametri di configurazione per l'applicazione. Questo file contiene tutti i parametri necessari per la distribuzione di CDK.
- `cdk`— Codice sorgente di AWS CDK.
- `setup.sh`— Lo script utilizzato per distribuire l'applicazione AWS CDK per creare la CI/CD pipeline richiesta e altre risorse richieste.
- `uninstall.sh`— Lo script utilizzato per ripulire le risorse.

Per utilizzare il codice di esempio, segui le istruzioni nella sezione Epics.

Best practice

Per le best practice per l'automazione di AWS Node Termination Handler, consulta quanto segue:

- Guide alle migliori pratiche EKS
- Node Termination Handler - Configurazione

Epiche

Configurare l'ambiente

Operazione	Description	Competenze richieste

Clona il repository.	Per clonare il repository utilizzando SSH (Secure Shell), esegui il seguente comando.
	`git clone git@github.com:aws-samples/deploy-nth-to-eks.git`
	Per clonare il repository utilizzando HTTPS, esegui il comando seguente.
	`git clone https://github.com/aws-samples/deploy-nth-to-eks.git`
	La clonazione del repository crea una cartella denominata. `deploy-nth-to-eks`
	Passa a quella directory.
		Sviluppatore di app, AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	`cd deploy-nth-to-eks`	
Imposta il file kubeconfig.	Imposta le tue credenzia li AWS nel tuo terminale e conferma di avere i diritti per assumere il ruolo di cluster. Puoi usare il seguente codice di esempio. `aws eks update-ku beconfig --name <Cluster_Name> --region <region>--role-arn <Role_ARN>`	AWS DevOps, DevOps ingegnere, sviluppatore di app

Implementa la pipeline CI/CD

Operazione	Description	Competenze richieste
Imposta i parametri.	Nel `config/config.json` file, impostate i seguenti parametri obbligatori. • `pipelineName` : il nome della CI/CD pipeline che deve essere creata da AWS CDK (ad esempio,`deploy-nth-to-eks-pipeline`). AWS CodePipeline creerà una pipeline con questo nome. • `repositoryName` : Il CodeCommit repositor	Sviluppatore di app, AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	y AWS da creare (ad esempio,`deploy-nth-to-eks-repo`). AWS CDK creerà questo repository e lo imposterà come origine per la CI/CD pipeline. **Note** Questa soluzione creerà questo CodeCommit repository e il ramo (fornito nel seguente parametro di ramo). • `branch`: Il nome del ramo nel repository (ad esempio,`main`). Un commit su questo ramo avvierà la CI/CD pipeline. • `cfn_scan_script` : il percorso dello script che verrà utilizzato per scansionare il CloudFormation modello AWS alla ricerca di NTH (`scan.sh`). Questo script esiste nella `nth` cartella che farà parte del CodeCommit repository AWS. • `cfn_deploy_script`: Il percorso dello script che verrà utilizzato per	

Operazione	Description	Competenze richieste
	distribuire il CloudForm ation modello AWS per NTH ()`installApp.sh` . • `stackName` : Il nome dello CloudFormation stack da distribuire. • `eksClusterName` : il nome del cluster EKS esistente. • `eksClusterRole` : Il ruolo IAM che verrà utilizzato per accedere al cluster EKS per tutte le chiamate API Kubernetes (ad esempio,). `clusteradmin` Di solito, questo ruolo viene aggiunto. `aws-auth ConfigMap` • `create_cluster_rol e` : Per creare il ruolo `eksClusterRole` IAM, inserisci yes. Se desideri fornire un ruolo cluster esistente nel `eksCluste rRole` parametro, inserisci no. • `create_iam_oidc_pr ovider` : Per creare il provider IAM OIDC per il tuo cluster, inserisci yes. Se esiste già un provider IAM OIDC, inserisci no. Per ulteriori informazioni, consulta Creazione di un	

Operazione	Description	Competenze richieste
	provider IAM OIDC per il cluster. • `AsgGroupName` : un elenco separato da virgole di nomi di gruppi Auto Scaling che fanno parte del cluster EKS (ad esempio,). `ASG_Group_1,ASG_Group_2` • `region`: il nome della regione AWS in cui si trova il cluster (ad esempio,`us-east-2`). • `install_cdk` : se AWS CDK non è attualmente installato sulla macchina, inserisci yes. Esegui il `cdk --version` comando per verificare se la versione di AWS CDK installata è 2.27.0 o successiva. In tal caso, inserisci no. Se inserisci yes, lo script setup.sh eseguirà il `sudo npm install -g cdk@2.27.0` comando per installare AWS CDK sulla macchina. Lo script richiede le autorizzazioni sudo, quindi fornisci la password dell'account quando richiesto.	

Operazione	Description	Competenze richieste
Crea la CI/CD pipeline per distribuire NTH.	Esegui lo script setup.sh. `./setup.sh` Lo script distribuirà l'applicazione AWS CDK che creerà il CodeCommit repository con codice di esempio, la pipeline e i CodeBuild progetti in base ai parametri di input dell'utente nel file. `config/config.json` Questo script richiederà la password mentre installa i pacchetti npm con il comando sudo.	Sviluppatore di app, AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
Esamina la CI/CD pipeline.	Apri la Console di gestione AWS ed esamina le seguenti risorse create nello stack. • CodeCommit repo con il contenuto della cartella `nth` • CodeBuild Progetto AWS`cfn-scan`, che analizzerà il CloudForm ation modello alla ricerca di vulnerabilità. • CodeBuild progetto`Nth-Deploy` , che distribuirà il CloudFormation modello AWS e i corrispondenti grafici NTH Helm attraverso la pipeline AWS. CodePipel ine • Una CodePipeline pipeline per implementare NTH. Dopo il corretto funzionam ento della pipeline, la versione Helm `aws-node-termination-handle r` viene installata nel cluster EKS. Inoltre, un Pod denominato `aws-node-termination-handler` è in esecuzione nello spazio dei `kube-system` nomi del cluster.	Sviluppatore di app, AWS DevOps, DevOps ingegnere

Prova l'implementazione di NTH

Operazione	Description	Competenze richieste
Simula un evento di scalabilità in gruppo Auto Scaling.	Per simulare un evento di scalabilità automatica, effettuate le seguenti operazioni: 1. Sulla console AWS, apri la EC2 console e scegli Auto Scaling Groups. 2. Seleziona il gruppo Auto Scaling con lo stesso nome di quello fornito in **`config/config.json`** e scegli Modifica. 3. Riduci la capacità desiderata e minima di 1. 4. Scegliere Aggiorna.	
Esamina i registri.	Durante l'evento di scale-in, l'NTH Pod collegherà e svuoterà il nodo di lavoro corrispondente (l' EC2 istanza che verrà terminata come parte dell'evento scale-in). Per controllare i log, usa il codice nella sezione Informazioni aggiuntive.	Sviluppatore di app, AWS DevOps, DevOps ingegnere

Eliminazione

Operazione	Description	Competenze richieste
Pulisci tutte le risorse AWS.	Per ripulire le risorse create da questo modello, esegui il comando seguente. `./uninstall.sh` Questo pulirà tutte le risorse create in questo modello eliminando lo CloudFormation stack.	DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
Il registro npm non è impostato correttamente.	Durante l'installazione di questa soluzione, lo script installa npm install per scaricare tutti i pacchetti richiesti. Se durante l'installazione viene visualizzato un messaggio che dice «Impossibile trovare il modulo», il registro npm potrebbe non essere impostato correttamente. Per visualizzare l'impostazione corrente del registro, esegui il comando seguente. `npm config get registry` Per impostare il registro con`https://registry.npmjs.org/` , esegui il comando seguente.

Problema	Soluzione
	`npm config set registry https://registry.npmjs.org`
Ritarda il recapito dei messaggi SQS.	Come parte della risoluzione dei problemi, se desideri ritardare la consegna dei messaggi SQS a NTH Pod, puoi modificare il parametro del ritardo di consegna di SQS. Per ulteriori informazioni, consulta le code di ritardo di Amazon SQS.

Risorse correlate

- Codice sorgente di AWS Node Termination Handler
- EC2 seminario
- AWS CodePipeline
- Amazon Elastic Kubernetes Service (Amazon EKS)
- Kit di sviluppo AWS per il cloud
- AWS CloudFormation

Informazioni aggiuntive

1. Trova il nome del Pod NTH.

```
kubectl get pods -n kube-system |grep aws-node-termination-handler
aws-node-termination-handler-65445555-kbqc7   1/1     Running   0          26m
kubectl get pods -n kube-system |grep aws-node-termination-handler
aws-node-termination-handler-65445555-kbqc7   1/1     Running   0          26m
```

2. Controllo dei log. Un registro di esempio è simile al seguente. Mostra che il nodo è stato isolato e drenato prima di inviare il segnale di completamento del ciclo di vita del gruppo Auto Scaling.

```
kubectl -n kube-system logs aws-node-termination-handler-65445555-kbqc7
022/07/17 20:20:43 INF Adding new event to the event store
 event={"AutoScalingGroupName":"eksctl-my-cluster-target-nodegroup-
```

```
ng-10d99c89-NodeGroup-ZME36IGAP7O1","Description":"ASG Lifecycle Termination
 event received. Instance will be interrupted at 2022-07-17 20:20:42.702
 +0000 UTC \n","EndTime":"0001-01-01T00:00:00Z","EventID":"asg-lifecycle-
term-33383831316538382d353564362d343332362d613931352d383430666165636334333564","InProgress":fal
east-2.compute.internal","NodeProcessed":false,"Pods":null,"ProviderID":"aws:///us-
east-2c/i-0409f2a9d3085b80e","StartTime":"2022-07-17T20:20:42.702Z","State":""}
2022/07/17 20:20:44 INF Requesting instance drain event-id=asg-lifecycle-
term-33383831316538382d353564362d343332362d613931352d383430666165636334333564
 instance-id=i-0409f2a9d3085b80e kind=SQS_TERMINATE node-name=ip-192-168-75-60.us-
east-2.compute.internal provider-id=aws:///us-east-2c/i-0409f2a9d3085b80e
2022/07/17 20:20:44 INF Pods on node node_name=ip-192-168-75-60.us-
east-2.compute.internal pod_names=["aws-node-qchsw","aws-node-termination-
handler-65445555-kbqc7","kube-proxy-mz5x5"]
2022/07/17 20:20:44 INF Draining the node
2022/07/17 20:20:44 ??? WARNING: ignoring DaemonSet-managed Pods: kube-system/aws-node-
qchsw, kube-system/kube-proxy-mz5x5
2022/07/17 20:20:44 INF Node successfully cordoned and drained
 node_name=ip-192-168-75-60.us-east-2.compute.internal reason="ASG Lifecycle
 Termination event received. Instance will be interrupted at 2022-07-17 20:20:42.702
 +0000 UTC \n"
2022/07/17 20:20:44 INF Completed ASG Lifecycle Hook (NTH-K8S-TERM-HOOK) for instance
 i-0409f2a9d3085b80e
```

Crea e distribuisci automaticamente un'applicazione Java su Amazon EKS utilizzando una pipeline CI/CD

MAHESH RAGHUNANDANAN, Jomcy Pappachen e James Radtke, Amazon Web Services

Riepilogo

Questo modello descrive come creare una pipeline di integrazione e distribuzione continua (CI/CD) che crea e distribuisce automaticamente un'applicazione Java con le DevSecOps pratiche consigliate in un cluster Amazon Elastic Kubernetes Service (Amazon EKS) su. Cloud AWS Questo modello utilizza un'applicazione di saluto sviluppata con un framework Java Spring Boot e che utilizza Apache Maven.

Puoi utilizzare l'approccio di questo pattern per creare il codice per un'applicazione Java, impacchettare gli artefatti dell'applicazione come immagine Docker, eseguire la scansione di sicurezza dell'immagine e caricare l'immagine come contenitore di carichi di lavoro su Amazon EKS. L'approccio di questo pattern è utile se desideri migrare da un'architettura monolitica strettamente accoppiata a un'architettura di microservizi. L'approccio consente inoltre di monitorare e gestire l'intero ciclo di vita di un'applicazione Java, garantendo un livello di automazione più elevato e contribuendo a evitare errori o bug.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- AWS Command Line Interface (AWS CLI) versione 2, installata e configurata. Per ulteriori informazioni su questo argomento, vedere Installazione o aggiornamento alla versione più recente di AWS CLI nella AWS CLI documentazione.

  AWS CLI la versione 2 deve essere configurata con lo stesso ruolo AWS Identity and Access Management (IAM) che crea il cluster Amazon EKS, poiché solo quel ruolo è autorizzato ad aggiungere altri ruoli IAM a `aws-authConfigMap`. Per informazioni e passaggi di configurazione AWS CLI, consulta Configurazione delle impostazioni nella AWS CLI documentazione.
- Ruoli e autorizzazioni IAM con accesso completo a. AWS CloudFormation Per ulteriori informazioni su questo argomento, consulta Controlling access with IAM nella CloudFormation documentazione.
- Un cluster Amazon EKS esistente, con dettagli sul nome del ruolo IAM e sull'Amazon Resource Name (ARN) del ruolo IAM per i nodi di lavoro nel cluster EKS.

- Kubernetes Cluster Autoscaler, installato e configurato nel tuo cluster Amazon EKS. Per ulteriori informazioni, consulta Scale cluster compute with Karpenter e Cluster Autoscaler nella documentazione di Amazon EKS.
- Accesso al codice nel repository. GitHub

> **Important**
>
> AWS Security Hub CSPM è abilitato come parte dei CloudFormation modelli inclusi nel codice di questo modello. Per impostazione predefinita, dopo l'attivazione di Security Hub CSPM, viene fornita una prova gratuita di 30 giorni. Dopo il periodo di prova, è previsto un costo. Servizio AWS Per ulteriori informazioni sui prezzi, consulta Prezzi di AWS Security Hub CSPM.

Versioni del prodotto

- Helm versione 3.4.2 o successiva
- Apache Maven versione 3.6.3 o successiva
- BridgeCrew Checkov versione 2.2 o successiva
- Aqua Security Trivy versione 0.37 o successiva

Architecture

Stack tecnologico

- AWS CodeBuild
- AWS CodeCommit
- Amazon CodeGuru
- AWS CodePipeline
- Amazon Elastic Container Registry (Amazon ECR)
- Amazon EKS
- Amazon EventBridge
- AWS Security Hub CSPM
- Amazon Simple Notification Service (Amazon SNS)

Architettura Target

Il diagramma mostra il flusso di lavoro seguente:

1. Lo sviluppatore aggiorna il codice dell'applicazione Java nel ramo base del CodeCommit repository, che crea una pull request (PR).
2. Non appena il PR viene inviato, Amazon CodeGuru Reviewer esamina automaticamente il codice, lo analizza in base alle migliori pratiche per Java e fornisce consigli allo sviluppatore.
3. Dopo che il PR è stato unito al ramo base, viene creato un EventBridge evento Amazon.
4. L' EventBridge evento avvia la CodePipeline pipeline, che si avvia.
5. CodePipeline esegue la fase di CodeSecurity scansione (sicurezza continua).
6. AWS CodeBuild avvia il processo di scansione di sicurezza in cui i file Helm della distribuzione Dockerfile e Kubernetes vengono scansionati utilizzando Checkov e il codice sorgente dell'applicazione viene scansionato in base alle modifiche incrementali del codice. La scansione del codice sorgente dell'applicazione viene eseguita dal wrapper CLI (Command Line Interface) di CodeGuru Reviewer.

   > **Note**
   >
   > A partire dal 7 novembre 2025, non puoi creare nuove associazioni di repository in Amazon CodeGuru Reviewer. Per informazioni sui servizi con funzionalità simili a CodeGuru Reviewer, consulta la modifica della disponibilità di Amazon CodeGuru Reviewer nella documentazione di CodeGuru Reviewer.

7. Se la fase di scansione di sicurezza ha esito positivo, viene avviata la fase di compilazione (integrazione continua).
8. Nella fase di compilazione, CodeBuild crea l'artefatto, lo impacchetta in un'immagine Docker, analizza l'immagine alla ricerca di vulnerabilità di sicurezza utilizzando Aqua Security Trivy e archivia l'immagine in Amazon ECR.
9. Le vulnerabilità rilevate dalla fase 8 vengono caricate su Security Hub CSPM per ulteriori analisi da parte di sviluppatori o ingegneri. Security Hub CSPM fornisce una panoramica e consigli per correggere le vulnerabilità.
10. Le notifiche e-mail delle fasi sequenziali all'interno della CodePipeline pipeline vengono inviate tramite Amazon SNS.

11. Una volta completate le fasi di integrazione continua, CodePipeline entra nella fase Deploy (distribuzione continua).
12. L'immagine Docker viene distribuita su Amazon EKS come carico di lavoro container (pod) utilizzando i grafici Helm.
13. Il pod dell'applicazione è configurato con l'agente Amazon CodeGuru Profiler, che invia i dati di profilazione dell'applicazione (CPU, utilizzo dell'heap e latenza) a CodeGuru Profiler, che aiuta gli sviluppatori a comprendere il comportamento dell'applicazione.

Tools (Strumenti)

Servizi AWS

- CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante il loro ciclo di vita in tutte le regioni. Account AWS
- AWS CodeBuildè un servizio di compilazione completamente gestito che consente di compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per l'implementazione.
- AWS CodeCommitè un servizio di controllo delle versioni che consente di archiviare e gestire in modo privato gli archivi Git, senza dover gestire il proprio sistema di controllo del codice sorgente.
- Amazon CodeGuru Profiler raccoglie dati sulle prestazioni di runtime dalle tue applicazioni live e fornisce consigli che possono aiutarti a ottimizzare le prestazioni delle tue applicazioni.
- AWS CodePipelineti aiuta a modellare e configurare rapidamente le diverse fasi di un rilascio del software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire AWS Kubernetes senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti, tra cui AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altro modo. Account AWS
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.

- AWS Security Hub CSPMfornisce una visione completa dello stato di sicurezza su. AWS Inoltre, consente di verificare la conformità AWS dell'ambiente agli standard e alle best practice del settore della sicurezza.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri servizi

- Helm è un gestore di pacchetti open source per Kubernetes.
- Apache Maven è uno strumento di comprensione e gestione di progetti software.
- BridgeCrew Checkov è uno strumento statico di analisi del codice per la scansione dei file Infrastructure as Code (IaC) alla ricerca di configurazioni errate che potrebbero portare a problemi di sicurezza o conformità.
- Aqua Security Trivy è uno scanner completo per le vulnerabilità nelle immagini dei container, nei file system e negli archivi Git, oltre ai problemi di configurazione.

Codice

Il codice per questo pattern è disponibile nel repository. GitHub aws-codepipeline-devsecops-amazoneks

Best practice

- Questo modello segue le migliori pratiche di sicurezza IAM per applicare il principio del privilegio minimo per le entità IAM in tutte le fasi della soluzione. Se desideri estendere la soluzione con strumenti aggiuntivi Servizi AWS o di terze parti, ti consigliamo di consultare la sezione sull'applicazione delle autorizzazioni con privilegi minimi nella documentazione IAM.
- Se disponi di più applicazioni Java, ti consigliamo di creare pipeline separate CI/CD per ogni applicazione.
- Se disponi di un'applicazione monolitica, ti consigliamo di suddividerla in microservizi, ove possibile. I microservizi sono più flessibili, semplificano la distribuzione delle applicazioni come contenitori e offrono una migliore visibilità sulla creazione e sulla distribuzione complessive dell'applicazione.

Epiche

Configura l'ambiente

Operazione	Description	Competenze richieste
Clona il GitHub repository.	Per clonare il repository, esegui il comando seguente. `git clone https://github.com/aws-samples/aws-codepipeline-devsecops-amazoneks`	Sviluppatore di app, ingegnere DevOps
Crea un bucket S3 e carica il codice.	1. Accedi a Console di gestione AWS, apri la console Amazon S3, quindi crea un bucket S3 nel punto in Regione AWS cui intendi implementare questa soluzione. Per ulteriori informazioni, consulta Creazione di un bucket nella documentazione di Amazon S3. 2. Nel bucket S3, crea una cartella denominata. `code` 3. Vai al punto in cui hai clonato il repository. Per creare una versione compressa dell'intero codice con l'estensione.zip (`cicdstack.zip`) e convalidare il file.zip, esegui i seguenti comandi nell'ordine.	AWS DevOps, amministratore del cloud, DevOps ingegnere

Operazione	Description	Competenze richieste

	`cd aws-codepipeline-devsecops-amazoneks`
	`python -m zipfile -c cicdstack.zip *`
	`python -m zipfile -t cicdstack.zip`
	
	Note
	Se il `python` comando fallisce e indica che Python non è stato trovato, usa `python3` invece.
	
	4. Carica il `cicdstack.zip` file nella cartella di codice che hai creato in precedenza nel bucket S3.	

Operazione	Description	Competenze richieste
Crea uno CloudFormation stack.	1. Aprire la console CloudFormation e scegliere Create Stack (Crea stack). 2. In Specificare modello, scegli Carica un file modello, carica il `cf_templates/codecommit_ecr.yaml` file e quindi scegli Avanti. 3. In Specificare i dettagli dello stack, inserisci il nome dello stack, quindi fornisci i seguenti valori dei parametri di input: • `CodeCommitRepositoryBranchName` : Il nome del ramo in cui risiederà il codice (l'impostazione predefinita è) `main` • `CodeCommitRepositoryName` : Il nome del file CodeCommitrepository da creare • `CodeCommitRepositoryS3Bucket` : il nome del bucket S3 in cui è stata creata la cartella del codice • `CodeCommitRepositoryS3BucketObjKey` : `code/cicdstack.zip`	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	• `ECRRepositoryName` : il nome del repository Amazon ECR da creare 4. Scegli Avanti, utilizza le impostazioni predefinite per le opzioni Configure stack, quindi scegli Avanti. 5. Nella sezione Revisione , verifica i dettagli del modello e dello stack, quindi scegli Crea pila. Viene quindi creato lo stack, inclusi i repository CodeCommit e Amazon ECR. 6. Prendi nota dei nomi dei repository CodeCommit e di Amazon ECR, che saranno necessari per configurare la pipeline Java CI/CD .	
Convalida l'implementazione dello CloudFormation stack.	1. In Stacks sulla CloudFormation console, verifica lo stato dello CloudFormation stack che hai distribuito. Lo stato dello stack deve essere CREATE COMPLETE. 2. Dalla console, verifica che i repository CloudFormation e Amazon ECR siano stati forniti e siano pronti.	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
Eliminare il bucket S3.	Svuota ed elimina il bucket S3 che hai creato in precedenza. Per ulteriori informazioni, consulta Eliminazione di un bucket nella documentazione di Amazon S3.	AWS DevOps, DevOps ingegnere

Configura i grafici Helm

Operazione	Description	Competenze richieste

Configura i grafici Helm della tua applicazione Java.	1. Nella posizione in cui hai clonato il GitHub repository, accedi alla cartella. `helm_charts/aws-proserve-java-greeting` In questa cartella, il `values.dev.yaml` file contiene informazioni sulla configurazione delle risorse Kubernetes che puoi modificare per le distribuzioni di container in Amazon EKS. Aggiorna il parametro del repository Docker fornendo il tuo Account AWS ID e il nome Regione AWS del repository Amazon ECR.
	`image:`
	`  repository:`
	`  <account-id>.dkr.ecr.<region>.amazon`
		DevOps ingegnere

Operazione	Description	Competenze richieste

	aws.com/<app-ecr-repo-name>
	2. Il tipo di servizio del pod Java è impostato su. `LoadBalancer`
	`service:`
	`type: LoadBalancer`
	`port: 80`
	`targetPort: 8080`
	`path: /hello`
	`initialDelaySeconds: 60`
	`periodSeconds: 30`
	Per utilizzare un servizio diverso (ad esempio,`NodePort`), è possibile modificare questo parametro. Per ulteriori informazioni, consulta la documentazione di Kubernetes.
	3. Puoi attivare Kubernetes Horizontal Pod Autoscaler modificando il parametro in. `autoscaling enabled: true`
	`autoscaling:`
	`enabled: true`
	`minReplicas: 1`
	`maxReplicas: 100`
	`targetCPUUtilizationPercentage: 80`	

Operazione	Description	Competenze richieste
	<pre> # targetMem oryUtilizationPerc entage: 80</pre> 4. Puoi abilitare diverse funzionalità per i carichi di lavoro Kubernetes modificando i valori nel `values.<ENV>.yaml` file, dove si trova il tuo ambiente di sviluppo, produzione, UAT o <ENV> QA.	

Operazione	Description	Competenze richieste

Convalida i grafici Helm per errori di sintassi.	1. Dal terminale, verificate che Helm v3 sia installat o nella workstation locale eseguendo il comando seguente.
	`helm --version`
	Se Helm v3 non è installato, installalo.
	2. Nel terminale, vai alla directory Helm charts (`helm_charts/aws-pr oserve-java-greeti ng`) ed esegui il seguente comando.
	`helm lint . -f values.dev.yaml`
	Questo controllerà i grafici Helm per eventuali errori di sintassi.
		DevOps ingegnere

Configura la CI/CD pipeline Java

Operazione	Description	Competenze richieste
Crea la CI/CD pipeline.	1. Aprire la console CloudFormation e scegliere Create Stack (Crea stack). 2. In Specificare modello, scegliete Carica un file modello, caricate il	AWS DevOps

Operazione	Description	Competenze richieste
	`cf_templates/build_deployment.yaml` modello, quindi scegliete Avanti. 3. In Specificare i dettagli dello stack, specificate il nome dello stack, quindi fornite i seguenti valori per i parametri di input: • `CodeBranchName` : nome del ramo del CodeCommit repository in cui risiede il codice • `EKSClusterName` : nome del cluster EKS (non l'`EKSCluster` ID) • `EKSCodeBuildAppName` : Nome dell'app Helm chart () `aws-proserve-java-greeting` • `EKSWorkerNodeRoleARN` : ARN del ruolo IAM assegnato ai nodi di lavoro Amazon EKS • `EKSWorkerNodeRoleName` : nome del ruolo IAM assegnato ai nodi di lavoro Amazon EKS • `EcrDockerRepository` : nome del repository Amazon ECR in cui verranno archiviate le	

Operazione	Description	Competenze richieste
	immagini Docker del codice • `EmailRecipient` : Indirizzo e-mail a cui inviare le notifiche di build • `EnvType`: Ambiente (ad esempio, dev, test o prod) • `SourceRepoName` : nome del CodeCommit repository in cui risiede il codice 4. Scegli Next (Successivo). Utilizza le impostazioni predefinite in Configura le opzioni dello stack, quindi scegli Avanti. 5. Nella sezione Revisione , verifica i dettagli del CloudFormation modello e dello stack, quindi scegli Avanti. 6. Seleziona Crea stack. 7. Durante la distribuzione CloudFormation dello stack, il proprietario dell'indirizzo e-mail che hai fornito nei parametri riceverà un messaggio per iscriversi a un argomento SNS. Per abbonarsi ad Amazon SNS, il proprietario deve scegliere il link nel messaggio.	

Operazione	Description	Competenze richieste
	8. Dopo aver creato lo stack, apri la scheda Output dello stack, quindi registra il valore ARN per la chiave di output. `EksCodeBuildkubeRoleARN` Questo valore IAM ARN verrà richiesto in seguito, quando fornirai le autorizzazioni per il ruolo CodeBuild IAM per distribuire carichi di lavoro nel cluster Amazon EKS.	

Attiva l'integrazione tra Security Hub CSPM e Aqua Security

Operazione	Description	Competenze richieste
Attiva l'integrazione con Aqua Security.	Questo passaggio è necessario per caricare i risultati di vulnerabilità delle immagini Docker segnalati da Trivy su Security Hub CSPM. Poiché CloudFormation non supporta le integrazioni CSPM di Security Hub, questo processo deve essere eseguito manualmente. 1. Apri la AWS Security Hub CSPM console e vai a Integrazioni.	Amministratore, DevOps ingegnere di AWS

Operazione	Description	Competenze richieste
	2. Cerca Aqua Security e seleziona Aqua Security: Aqua Security. 3. Scegli Accetta risultati.	

Configura CodeBuild per eseguire i comandi Helm o kubectl

Operazione	Description	Competenze richieste
Consenti CodeBuild di eseguire comandi Helm o kubectl nel cluster Amazon EKS.	CodeBuild Per autentica rti per utilizzare Helm o `kubectl` i comandi con il cluster Amazon EKS, devi aggiungere i ruoli IAM a. `aws-auth ConfigMap` In questo caso, aggiungi l'ARN del ruolo IAM`EksCodeBu ildkubeRoleARN` , che è il ruolo IAM creato per consentir e al CodeBuild servizio di accedere al cluster Amazon EKS e distribuire carichi di lavoro su di esso. Questa è un'attività una tantum. **Important** La seguente procedura deve essere completat a prima della fase di approvazione della distribuzione. CodePipeline	DevOps

Operazione	Description	Competenze richieste

	1. Apri lo script della `cf_templates/kube_aws_auth_configmap_patch.sh` shell nel tuo ambiente Amazon Linux o macOS.
	2. Effettua l'autenticazione sul cluster Amazon EKS eseguendo il comando seguente.
	`aws eks --region <aws-region> update-kubeconfig --name <eks-cluster-name>`
	3. Esegui lo script della shell utilizzando il seguente comando, sostituendolo `<rolearn-eks-codebuild-kubectl>` con il valore ARN registrato `EksCodeBuildkubeRoleARN` in precedenza.
	`bash cf_templates/kube_aws_auth_configmap_patch.sh <rolearn-eks-codebuild-kubectl>`
	`aws_authConfigMap` è configurato e l'accesso è concesso.	

Convalida la pipeline CI/CD

Operazione	Description	Competenze richieste

Verificate che la CI/CD pipeline si avvii automatic amente.	1. La fase di CodeSecurity scansione nella pipeline di solito fallisce se Checkov rileva vulnerabilità nei grafici Dockerfile o Helm. Tuttavia, lo scopo di questo esempio è stabilire un processo di identific azione delle potenziali vulnerabilità di sicurezza anziché correggerle attraverso la pipeline, in genere un processo. CI/CD DevSecOps Nel file`buildspec/buildspec_secscan.yaml` , il `checkov` comando utilizza il `--soft-fail` flag per evitare errori nella pipeline.
	`- echo -e "\n Running Dockerfile Scan"`
	`- checkov -f code/app/Dockerfile --framework dockerfile --soft-fail --summary-position bottom`
	`- echo -e "\n Running Scan of Helm Chart files"`
	`- cp -pv helm_charts/$EKS_CODEBUILD_APP_NAME/values.dev.yaml helm_char`
		DevOps

Operazione	Description	Competenze richieste
	ts/$EKS_CODEBUILD_APP_NAME/values.yaml - checkov -d helm_charts/$EKS_CODEBUILD_APP_NAME --framework helm --soft-fail --summary-position bottom - rm -rfv helm_charts/$EKS_CODEBUILD_APP_NAME/values.yaml Affinché la pipeline fallisca quando vengono segnalate vulnerabilità per i grafici Dockerfile e Helm, è necessario rimuovere l'`--soft-fail` opzione dal comando. `checkov` Gli sviluppatori o gli ingegneri possono quindi correggere le vulnerabilità e inserire le modifiche nell'archivio del codice sorgente. CodeCommit 2. Analogamente a CodeSecurity Scan, la fase di compilazione utilizza Aqua Security Trivy per identificare le `HIGH` vulnerabilità delle immagini `CRITICAL` Docker prima di inviare l'applicazione ad Amazon ECR.	

Operazione	Description	Competenze richieste

	<pre>- AWS_REGION=$AWS_DEFAULT_REGION
	 AWS_ACCOUNT_ID=$AWS_ACCOUNT_ID trivy -d image --no-progress --ignore-unfixed --exit-code 0 --severity HIGH,CRITICAL --format template --template "@securityhub/asff.tpl" -o securityhub/report.asff $AWS_ACCOUNT_ID.dkr.ecr.$AWS_DEFAULT_REGION.amazonaws.com/$IMAGE_REPO_NAME:$CODEBUILD_RESOLVED_SOURCE_VERSION</pre>
	In questo esempio, la pipeline non fallisce quando vengono segnalate vulnerabilità dell'immagine Docker, perché il `trivy` comando nel file include il flag con un valore di. `buildspec/buildspec.yml` **`--exit-code 0`** Affinché la pipeline fallisca quando vengono `HIGH` segnalate `CRTICAL` vulnerabilità, modifica il valore di to. `--exit-code 1` Gli sviluppatori o gli ingegneri possono quindi corregger	

Operazione	Description	Competenze richieste
	e le vulnerabilità e inserire le modifiche nell'archivio del CodeCommit codice sorgente. 3. Le vulnerabilità delle immagini Docker segnalate da Aqua Security Trivy vengono caricate su Security Hub CSPM. Nella console Security Hub CSPM, accedi a Findings. Filtra i risultati con Record State = Active e Product = Aqua Security. Questo elenca le vulnerabilità delle immagini Docker nel Security Hub CSPM. Possono essere necessari da 15 minuti a un'ora prima che le vulnerabilità compaiano in Security Hub CSPM. Per ulteriori informazioni sull'avvio della pipeline utilizzando CodePipeline, consulta Avviare una pipeline inCodePipeline, Avviare una pipeline manualmente e Avviare una pipeline in base a una pianificazione nella documentazione. CodePipeline	

Operazione	Description	Competenze richieste
Approvare la distribuzione.	1. Una volta completata la fase di compilazione, c'è una porta di approvazione della distribuzione. Il revisore o un release manager devono ispezionare la build e, se tutti i requisiti sono soddisfatti, approvarla. Questo è l'approccio consigliato per i team che utilizzano la distribuzione continua per la distribuzione delle applicazioni. 2. Dopo l'approvazione, la pipeline avvia la fase di distribuzione. 3. Una volta completata la fase di distribuzione, il CodeBuild registro di questa fase fornisce l'URL dell'applicazione. Utilizzate l'URL per convalidare la disponibilità dell'applicazione.	DevOps

Operazione	Description	Competenze richieste
Convalida la profilazione dell'applicazione.	Una volta completata la distribuzione e distribuito il pod dell'applicazione in Amazon EKS, l'agente Amazon CodeGuru Profiler configura to nell'applicazione proverà a inviare i dati di profilazi one dell'applicazione (CPU, riepilogo dell'heap, latenza e colli di bottiglia) a Profiler. CodeGuru Per la distribuzione iniziale di un'applicazione, CodeGuru Profiler impiega circa 15 minuti per visualizzare i dati di profilazione.	AWS DevOps

Risorse correlate

- AWS CodePipeline documentazione
- Scansione di immagini con Trivy in un AWS CodePipeline (AWS post sul blog)
- Miglioramento delle applicazioni Java con Amazon CodeGuru Profiler (post di AWS blog)
- AWS Sintassi del Security Finding Format (ASFF)
- Modelli di EventBridge eventi Amazon
- Aggiornamento del timone

Informazioni aggiuntive

- CodeGuru Profiler non deve essere confuso con il AWS X-Ray servizio in termini di funzionalità. Ti consigliamo di utilizzare CodeGuru Profiler per identificare le righe di codice più costose che potrebbero causare strozzature o problemi di sicurezza e risolverle prima che diventino un potenziale rischio. Il servizio X-Ray serve per il monitoraggio delle prestazioni delle applicazioni.

- In questo modello, le regole degli eventi sono associate al bus di eventi predefinito. Se necessario, è possibile estendere il pattern per utilizzare un bus di eventi personalizzato.
- Questo modello utilizza CodeGuru Reviewer come strumento statico di test della sicurezza delle applicazioni (SAST) per il codice dell'applicazione. Puoi utilizzare questa pipeline anche per altri strumenti, come SonarQube Checkmarx. È possibile aggiungere le istruzioni di configurazione della scansione per ognuno di questi strumenti in sostituzione delle CodeGuru istruzioni `buildspec/buildspec_secscan.yaml` di scansione.

> **Note**
>
> A partire dal 7 novembre 2025, non puoi creare nuove associazioni di repository in Amazon CodeGuru Reviewer. Per informazioni sui servizi con funzionalità simili a CodeGuru Reviewer, consulta la modifica della disponibilità di Amazon CodeGuru Reviewer nella documentazione di CodeGuru Reviewer.

Copia le immagini dei container Amazon ECR su e Account AWS Regioni AWS

Faisal Shahdad, Amazon Web Services

Riepilogo

Questo modello mostra come utilizzare un approccio serverless per replicare immagini con tag da repository Amazon Elastic Container Registry (Amazon ECR) esistenti su altri e. Account AWS Regioni AWS La soluzione consente di gestire il flusso AWS Step Functions di lavoro di replica e le AWS Lambda funzioni per copiare immagini di contenitori di grandi dimensioni.

Amazon ECR utilizza funzionalità di replica native tra regioni e account che replicano le immagini dei container tra regioni e account. Tuttavia, queste funzionalità replicano le immagini solo dal momento in cui la replica viene attivata. Non esiste alcun meccanismo per replicare le immagini esistenti in diverse regioni e account.

Questo modello aiuta i team di intelligenza artificiale (AI) a distribuire modelli, framework (ad esempio, e TensorFlow Hugging Face) e dipendenze di machine learning (ML) containerizzati ad altri account e regioni. PyTorch Questo può aiutarti a superare i limiti del servizio e ottimizzare le risorse di calcolo della GPU. Puoi anche replicare selettivamente i repository Amazon ECR da account e regioni di origine specifici. Per ulteriori informazioni, consulta la pagina Cross-Region Replication in Amazon ECR has landed.

Prerequisiti e limitazioni

Prerequisiti

- Due o più account attivi Account AWS (almeno un account di origine e un account di destinazione)
- Autorizzazioni appropriate AWS Identity and Access Management (IAM) in tutti gli account
- Docker per creare l'immagine del contenitore Lambda
- AWS Command Line Interface (AWS CLI) configurato per tutti gli account

Limitazioni

- Esclusione di immagini senza tag: la soluzione copia solo le immagini dei contenitori con tag espliciti. Ignora le immagini senza tag che esistono con i digest. `SHA256`

- Vincoli di timeout di esecuzione Lambda: AWS Lambda è limitato a un timeout di esecuzione massimo di 15 minuti, che potrebbe essere insufficiente per copiare immagini di contenitori o repository di grandi dimensioni.
- Gestione manuale dell'immagine del contenitore: il codice `crane-app.py` Python richiede la ricostruzione e la ridistribuzione dell'immagine del contenitore Lambda.
- Capacità di elaborazione parallela limitata: l'impostazione `MaxConcurrency` dello stato limita il numero di repository che è possibile copiare contemporaneamente. Tuttavia, puoi modificare questa impostazione nel AWS CloudFormation modello dell'account di origine. Tieni presente che valori di concorrenza più elevati possono causare il superamento dei limiti di velocità di servizio e delle quote di esecuzione Lambda a livello di account.

Architecture

Pila di obiettivi

Il pattern ha quattro componenti principali:

- Infrastruttura dell'account di origine: CloudFormation modello che crea i componenti di orchestrazione
- Infrastruttura dell'account di destinazione: CloudFormation modello che crea ruoli di accesso tra account
- Funzione Lambda: funzione basata su Python che utilizza Crane per una copia efficiente delle immagini
- Immagine del contenitore: contenitore Docker che racchiude la funzione Lambda con gli strumenti richiesti

Architettura Target

Flusso di lavoro Step Functions

La macchina a stati Step Functions orchestra quanto segue, come mostrato nel diagramma seguente:

- `PopulateRepositoryList`— Scansiona i repository Amazon ECR e popola Amazon DynamoDB
- `GetRepositoryList`— Recupera un elenco di repository univoco da DynamoDB

- `DeduplicateRepositories`— Assicura che non vi siano elaborazioni duplicate
- `CopyRepositories`— Gestisce la copia parallela dei repository
- `NotifySuccess`/`NotifyFailure`— Notifiche Amazon Simple Notification Service (Amazon SNS) basate sul risultato dell'esecuzione

Tools (Strumenti)

Strumenti Amazon

- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue AWS risorse e delle applicazioni su cui esegui AWS in tempo reale.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS Step Functionsè un servizio di orchestrazione serverless che consente di combinare funzioni Lambda e altro Servizi AWS per creare applicazioni aziendali critiche.

Altri strumenti

- Crane è uno strumento di orchestrazione Docker. È simile a Docker Compose ma ha funzionalità aggiuntive.
- Docker è un insieme di prodotti Platform as a Service (PaaS) che utilizzano la virtualizzazione a livello di sistema operativo per fornire software in contenitori.

Archivio di codici

- Il codice per questo pattern è disponibile nel GitHub sample-ecr-copy repository. È possibile utilizzare il CloudFormation modello del repository per creare le risorse sottostanti.

Best practice

Segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.

Epiche

Prepara un ambiente

Operazione	Description	Competenze richieste

Configura AWS CLI i profili.	1. Configura il profilo dell'account di origine:
	`aws configure --profile source-account`
	`# Enter: Access Key ID, Secret Access Key, Default region, Output format (json)`
	2. Configura il profilo dell'account di destinazione:
	`aws configure --profile destination-account`
	`# Enter: Access Key ID, Secret Access Key, Default region, Output format (json)`
	3. Verifica le configurazioni:
	`aws sts get-caller-identity --profile source-account`
		DevOps ingegnere, ingegnere dei dati, ingegnere ML

Operazione	Description	Competenze richieste
	`aws sts get-caller-identity --profile destination-account`	

Operazione	Description	Competenze richieste

Raccogli le informazioni richieste.	1. Ottieni l'ID dell'account di origine:
	`export SOURCE_ACCOUNT_ID=$(aws sts get-caller-identity --profile source-account --query Account --output text)`
	`echo "Source Account ID: $SOURCE_ACCOUNT_ID"`
	2. Ottieni l'ID dell'account di destinazione:
	`export DEST_ACCOUNT_ID=$(aws sts get-caller-identity --profile destination-account --query Account --output text)`
	`echo "Destination Account ID: $DEST_ACCOUNT_ID"`
	3. Imposta il Regioni AWS. Modifica questo comando per la tua regione:
	`export SOURCE_REGION="us-east-1"`
	`export DEST_REGION="us-east-2"`
	4. Elenca i repository Amazon ECR esistenti nell'account di origine:
		DevOps ingegnere, ingegnere dati, ingegnere ML

Operazione	Description	Competenze richieste
	`aws ecr describe-repositories --profile source-account --region $SOURCE_REGION --query 'repositories[].repositoryName' --output table`	
Clonare il repository.	Clona il repository del pattern sulla tua workstation locale: `git clone https://github.com/aws-samples/sample-ecr-copy`	DevOps ingegnere, ingegnere dei dati, ingegnere ML

Implementa l'infrastruttura per l'account di destinazione

Operazione	Description	Competenze richieste
Convalida il modello.	Convalida il modello: CloudFormation `aws cloudformation validate-template \ --template-body file://"Destination Account cf_template.yml" \ --profile destination-account`	DevOps ingegnere, ingegnere ML, ingegnere dati
Implementa l'infrastruttura di destinazione.	1. Distribuisci lo stack di account di destinazione:	Ingegnere dei dati, ingegnere ML, DevOps ingegnere

Operazione	Description	Competenze richieste

	``
	aws cloudformation
	 deploy \
	  --template-file
	 "Destination Account
	 cf_template.yml" \
	  --stack-name ecr-
	copy-destination \
	  --parameter-overri
	des \
	  SourceAccountId=
	$SOURCE_ACCOUNT_ID \
	  SourceRol
	eName=ECRContainer
	LambdaRole \
	  --capabilities
	 CAPABILITY_NAMED_I
	AM \
	  --profile destinati
	on-account \
	  --region $DEST_REG
	ION
	``
	2. Attendi il completamento dello stack:
	``
	aws cloudformation
	 wait stack-create-
	complete \
	  --stack-name ecr-
	copy-destination \
	  --profile destinati
	on-account \
	  --region $DEST_REG
	ION
	``	

Operazione	Description	Competenze richieste

Verifica l'implementazione.	1. Ottieni gli output dello stack:
	`aws cloudformation describe-stacks \ --stack-name ecr-copy-destination \ --profile destination-account \ --region $DEST_REGION \ --query 'Stacks[0].Outputs' \ --output table`
	2. Memorizza il ruolo IAM su più account:
	`export CROSS_ACCOUNT_ROLE_ARN=$(aws cloudformation describe-stacks \ --stack-name ecr-copy-destination \ --profile destination-account \ --region $DEST_REGION \ --query 'Stacks[0].Outputs[?OutputKey==`CrossAccountRoleArn`].OutputValue' \ --output text)`
	`echo "Cross-Account Role ARN: $CROSS_ACCOUNT_ROLE_ARN"`
		DevOps ingegnere, ingegnere ML, ingegnere dati

Crea e distribuisci l'immagine del contenitore Lambda

Operazione	Description	Competenze richieste

Prepara la costruzione del contenitore.	1. Verifica che Docker sia in esecuzione:
	`docker --version`
	`docker info`
	2. Assicurati che `crane-app.py` e `Dockerfile` si trovino nella directory corrente:
	`ls -la crane-app.py Dockerfile`
		Ingegnere dei dati, ingegnere ML, DevOps ingegnere

Crea l'immagine del contenitore.	1. Crea l'immagine del contenitore Lambda:
	`docker build -t ecr-copy-lambda . --no-cache`
	2. Verifica che l'immagine sia stata creata:
	`docker images ecr-copy-lambda`
	3. (Facoltativo) Prova il contenitore localmente:
	`docker run --rm --entrypoint python ecr-copy-lambda -c "import boto3;`
		Ingegnere dei dati, ingegnere ML, DevOps ingegnere

Operazione	Description	Competenze richieste
	`print('Container working')"`	

Operazione	Description	Competenze richieste

Crea un repository e carica l'immagine.	1. Crea un repository Amazon ECR nell'account di origine:
	`aws ecr create-repository \ --repository-name ecr-copy-lambda \ --profile source-account \ --region $SOURCE_REGION`
	2. Ottieni un token di accesso Amazon ECR e autentica Docker:
	`aws ecr get-login-password \ --profile source-account \ --region $SOURCE_REGION | \ docker login --username AWS --password-stdin \ $SOURCE_ACCOUNT_ID.dkr.ecr.$SOURCE_REGION.amazonaws.com`
	3. Etichetta l'immagine per Amazon ECR:
	`docker tag ecr-copy-lambda:latest \ $SOURCE_ACCOUNT_ID.dkr.ecr.$SOURCE_REGION.amazonaws.com/ecr-copy-lambda:latest`
		Ingegnere dei dati, ingegnere ML, DevOps ingegnere

Operazione	Description	Competenze richieste

	4. Carica l'immagine su Amazon ECR:
	
	`docker push $SOURCE_ACCOUNT_ID.dkr.ecr.$SOURCE_REGION.amazonaws.com/ecr-copy-lambda:latest`
	
	5. Memorizza l'URI dell'immagine per un uso successivo:
	
	`export LAMBDA_IMAGE_URI="$SOURCE_ACCOUNT_ID.dkr.ecr.$SOURCE_REGION.amazonaws.com/ecr-copy-lambda:latest"`
	`echo "Lambda Image URI: $LAMBDA_IMAGE_URI"`	

Operazione	Description	Competenze richieste

Verifica l'immagine.	1. Elenca le immagini nel repository:
	`aws ecr list-images \ --repository-name ecr-copy-lambda \ --profile source-account \ --region $SOURCE_REGION`
	2. Ottieni i dettagli dell'immagine:
	`aws ecr describe-images \ --repository-name ecr-copy-lambda \ --profile source-account \ --region $SOURCE_REGION`
		Ingegnere dei dati, ingegnere ML, DevOps ingegnere

Implementa l'infrastruttura dell'account di origine

Operazione	Description	Competenze richieste
Preparare i parametri di distribuzione.	1. Imposta l'e-mail di notifica: `export NOTIFICATION_EMAIL="your-email@company.com"` 2. Definisci i repository da copiare (separati da virgole):	Ingegnere dei dati, DevOps ingegnere, ingegnere ML

Operazione	Description	Competenze richieste

```
export REPOSITOR
Y_LIST="app-fronte
nd,app-backend,dat
abase-migrations"
```

3. Imposta l'ambiente:

```
export ENVIRONME
NT="dev"

echo "Deployment
 Parameters:"
echo "Source Account:
 $SOURCE_ACCOUNT_ID"
echo "Destination
 Account: $DEST_ACC
OUNT_ID"
echo "Source Region:
 $SOURCE_REGION"
echo "Destination
 Region: $DEST_REG
ION"
echo "Lambda Image:
 $LAMBDA_IMAGE_URI"
echo "Notification
 Email: $NOTIFICA
TION_EMAIL"
echo "Repositories:
 $REPOSITORY_LIST"
```

Operazione	Description	Competenze richieste
Convalida il modello sorgente.	Convalida il modello sorgente: CloudFormation `aws cloudformation validate-template \ --template-body file://"Source Account Cf template.yml" \ --profile source-account`	Ingegnere dei dati, ingegnere ML, DevOps ingegnere

Operazione	Description	Competenze richieste
Implementa l'infrastruttura di origine.	1. Implementa lo stack di account di origine: `aws cloudformation deploy \ --template-file "Source Account Cf template.yml" \ --stack-name ecr-copy-source \ --parameter-overrides \ SourceAccountId=$SOURCE_ACCOUNT_ID \ DestinationAccountId=$DEST_ACCOUNT_ID \ DestinationRegion=$DEST_REGION \ SourceRegion=$SOURCE_REGION \ NotificationEmail=$NOTIFICATION_EMAIL \ RepositoryList="$REPOSITORY_LIST" \ LambdaImageUri=$LAMBDA_IMAGE_URI \ Environment=$ENVIRONMENT \ --capabilities CAPABILITY_NAMED_IAM \ --profile source-account \ --region $SOURCE_REGION`	Ingegnere dei dati, ingegnere ML, DevOps ingegnere

Operazione	Description	Competenze richieste
	2. Attendi il completamento dello stack (questa operazione potrebbe richiedere fino a 10 minuti): `aws cloudformation wait stack-create-complete \ --stack-name ecr-copy-source \ --profile source-account \ --region $SOURCE_REGION`	

Operazione	Description	Competenze richieste

Verifica l'implementazione e raccogli gli output.	1. Ottieni gli output dello stack:
	`aws cloudformation describe-stacks \ --stack-name ecr-copy-source \ --profile source-account \ --region $SOURCE_REGION \ --query 'Stacks[0].Outputs' \ --output table`
	2. Memorizza Amazon Resource Names (ARNs) per la macchina a stati e l'argomento SNS:
	`export STATE_MACHINE_ARN=$(aws cloudformation describe-stacks \ --stack-name ecr-copy-source \ --profile source-account \ --region $SOURCE_REGION \ --query 'Stacks[0].Outputs[?OutputKey==`StateMachineArn`].OutputValue' \ --output text)`
	`export SNS_TOPIC_ARN=$(aws cloudformation describe-stacks \`
		DevOps ingegnere, ingegnere ML, ingegnere dati

Operazione	Description	Competenze richieste

	`--stack-name ecr-copy-source \ --profile source-account \ --region $SOURCE_REGION \ --query 'Stacks[0].Outputs[?OutputKey==`SNSTopicArn`].OutputValue' \ --output text)`
	
	`echo "State Machine ARN: $STATE_MACHINE_ARN" echo "SNS Topic ARN: $SNS_TOPIC_ARN"`	

Conferma la tua iscrizione via e-mail.	1. Controlla la tua e-mail per la conferma dell'abbonamento a SNS. 2. Scegli il link di conferma nell'e-mail. 3. Verifica lo stato dell'abbonamento. `aws sns list-subscriptions-by-topic \ --topic-arn $SNS_TOPIC_ARN \ --profile source-account \ --region $SOURCE_REGION`	Ingegnere dei dati, ingegnere ML, DevOps ingegnere

Esegui e monitora il processo di copia

Operazione	Description	Competenze richieste
Esegui e monitora il processo di copia.	1. Accedi a e apri la console Step Functions. Console di gestione AWS 2. Individua la macchina a stati. 3. Selezionare Start execution (Avvia esecuzione). Al termine, i risultati vengono visualizzati nella scheda Esecuzione input e output. 4. (Facoltativo) Se vuoi continuare a eseguire Step Functions utilizzando il AWS CLI, segui i passaggi rimanenti di questa epopea.	DevOps ingegnere, ingegnere ML, ingegnere dati

Esegui la funzione step.	1. Genera un nome univoco:
	
	`export EXECUTION_NAME="ecr-copy-$(date +%Y%m%d-%H%M%S)"`
	
	2. Esegui la funzione step.
	
	`export EXECUTION_ARN=$(aws stepfunctions start-execution \ --state-machine-arn $STATE_MACHINE_ARN \ --name $EXECUTION_NAME \`
		DevOps ingegnere, ingegnere ML, ingegnere dati

Operazione	Description	Competenze richieste

	<pre> --profile source-account \
	  --region $SOURCE_REGION \
	  --query 'executionArn' \
	  --output text)
	
	echo "Execution started: $EXECUTION_ARN"
	echo "Execution Name: $EXECUTION_NAME"</pre>	

Operazione	Description	Competenze richieste

Monitora i progressi.	1. Controlla lo stato:
	`aws stepfunctions describe-execution \ --execution-arn $EXECUTION_ARN \ --profile source-account \ --region $SOURCE_REGION \ --query '{Status:status,StartDate:startDate,StopDate:stopDate}' \ --output table`
	2. Ottieni la cronologia:
	`aws stepfunctions get-execution-history \ --execution-arn $EXECUTION_ARN \ --profile source-account \ --region $SOURCE_REGION \ --query 'events[?type==`TaskStateEntered` || type==`TaskSucceeded` || type==`TaskFailed`].{Type:type,Timestamp:timestamp,Details:stateEnteredEventDetails.name}' \ --output table`
		DevOps ingegnere, ingegnere ML, ingegnere dati

Operazione	Description	Competenze richieste

Controlla i risultati.	Attendi il completamento del processo (aggiornato ogni 30 secondi):
	
	`while true; do`
	`  STATUS=$(aws`
	` stepfunctions`
	` describe-execution \`
	`    --execution-arn`
	` $EXECUTION_ARN \`
	`    --profile source-ac`
	`count \`
	`    --region $SOURCE_R`
	`EGION \`
	`    --query 'status' \`
	`    --output text)`
	
	`  echo "Current status:`
	` $STATUS"`
	
	`  if [[ "$STATUS"`
	` == "SUCCEEDED"`
	` || "$STATUS" ==`
	` "FAILED" || "$STATUS"`
	` == "TIMED_OUT"`
	` || "$STATUS" ==`
	` "ABORTED" ]]; then`
	`    break`
	`  fi`
	
	`  sleep 30`
	`done`
	
	`echo "Final execution`
	` status: $STATUS"`
		DevOps ingegnere, ingegnere ML, ingegnere dati

Operazione	Description	Competenze richieste

Verifica le immagini.	1. Elenca i repository nell'account di destinazione:
	`aws ecr describe-repositories \ --profile destination-account \ --region $DEST_REGION \ --query 'repositories[].repositoryName' \ --output table`
	2. Controlla le immagini del repository:
	`for repo in $(echo $REPOSITORY_LIST | tr ',' ' '); do echo "\nImages in repository: $repo" aws ecr list-images \ --repository-name $repo \ --profile destination-account \ --region $DEST_REGION \ --query 'imageIds[].imageTag' \ --output table 2>/dev/null || echo "Repository $repo not found or no images" done`
		DevOps ingegnere, ingegnere dei dati, ingegnere ML

risoluzione dei problemi

Problema	Soluzione

Le funzioni Step non vengono eseguite.	1. Per recuperare gli eventi di errore dettagliati dalla cronologia, esegui il AWS CLI comando seguente:
	
	`if [[ "$STATUS" == "FAILED" ]]; then`
	`  echo "Getting failure details..."`
	`  aws stepfunctions get-execution-history \`
	`  --execution-arn $EXECUTION_ARN \`
	`  --profile source-account \`
	`  --region $SOURCE_REGION \`
	`  --query 'events[?type==`TaskFailed`]' \`
	`  --output json`
	`fi`
	
	2. Per recuperare i log delle funzioni Lambda non riuscite, esegui il seguente comando: AWS CLI
	
	`# Check Lambda function logs`
	`echo "\nLambda function logs:"`
	`aws logs describe-log-groups \`
	`  --log-group-name-prefix "/aws/lambda/ecr-copy-source" \`
	`  --profile source-account \`
	`  --region $SOURCE_REGION \`
	`  --query 'logGroups[].logGroupName' \`
	`  --output table`

Risorse correlate

- Documentazione Crane

- Cos'è Amazon Elastic Container Registry?
- Che cos'è AWS Lambda?
- Che cos'è Step Functions?

Informazioni aggiuntive

Parametri di configurazione

Parametro	Description	Esempio
`SourceAccountId`	Account AWS ID di origine	`11111111111`
`DestinationAccountId`	Account AWS ID di destinazione	`22222222222`
`DestinationRegion`	Obiettivo Regione AWS	`us-east-2`
`SourceRegion`	Fonte Regione AWS	`us-east-1`
`NotificationEmail`	Email per le notifiche	`abc@xyz.com`
`RepositoryList`	Archivi da copiare	`repo1,repo2,repo3`
`LambdaImageUri`	URI dell'immagine del contenitore Lambda	`${ACCOUNT}.dkr.ecr.${REGION}.amazonaws.com/ecr-copy-lambda:latest`

Crea una definizione di attività Amazon ECS e monta un file system su EC2 istanze utilizzando Amazon EFS

Durga Prasad Cheepuri, Amazon Web Services

Riepilogo

Questo modello fornisce esempi di codice e passaggi per creare una definizione di attività Amazon Elastic Container Service (Amazon ECS) da eseguire su istanze Amazon Elastic Compute Cloud (EC2Amazon) nel cloud Amazon Web Services (AWS), utilizzando Amazon Elastic File System (Amazon EFS) per montare un file system su tali istanze. EC2 Le attività di Amazon ECS che utilizzano Amazon EFS montano automaticamente i file system specificati nella definizione dell'attività e li rendono disponibili per i contenitori dell'attività in tutte le zone di disponibilità in una regione AWS.

Per soddisfare i tuoi requisiti di storage persistente e di storage condiviso, puoi usare Amazon ECS e Amazon EFS insieme. Ad esempio, puoi utilizzare Amazon EFS per archiviare dati utente persistenti e dati applicativi per le tue applicazioni con coppie di contenitori ECS attivi e in standby in esecuzione in diverse zone di disponibilità per un'elevata disponibilità. Puoi anche utilizzare Amazon EFS per archiviare dati condivisi a cui è possibile accedere in parallelo dai container ECS e dai carichi di lavoro distribuiti.

Per utilizzare Amazon EFS con Amazon ECS, puoi aggiungere una o più definizioni di volume a una definizione di attività. Una definizione di volume include un ID del file system Amazon EFS, un ID del punto di accesso e una configurazione per l'autorizzazione AWS Identity and Access Management (IAM) o la crittografia Transport Layer Security (TLS) in transito. È possibile utilizzare le definizioni dei contenitori all'interno delle definizioni delle attività per specificare i volumi di definizione delle attività che vengono montati durante l'esecuzione del contenitore. Quando viene eseguita un'attività che utilizza un file system Amazon EFS, Amazon ECS garantisce che il file system sia montato e disponibile per i contenitori che devono accedervi.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un cloud privato virtuale (VPC) con un endpoint o un router di rete privata virtuale (VPN)
- (Consigliato) Agente container Amazon ECS 1.38.0 o versione successiva per la compatibilità con i punti di accesso Amazon EFS e le funzionalità di autorizzazione IAM (per ulteriori informazioni, consulta il post del blog AWS New for Amazon EFS — IAM Authorization and Access Points).

Limitazioni

- Le versioni di Amazon ECS Container Agent precedenti alla 1.35.0 non supportano i file system Amazon EFS per le attività che utilizzano il EC2 tipo di avvio.

Architecture

Il diagramma seguente mostra un esempio di applicazione che utilizza Amazon ECS per creare una definizione di attività e montare un file system Amazon EFS su EC2 istanze in contenitori ECS.

Il diagramma mostra il flusso di lavoro seguente:

1. Crea un file system Amazon EFS.
2. Crea una definizione di attività con un contenitore.
3. Configura le istanze del contenitore per montare il file system Amazon EFS. La definizione del task fa riferimento ai montaggi del volume, quindi l'istanza del contenitore può utilizzare il file system Amazon EFS. Le attività ECS hanno accesso allo stesso file system Amazon EFS, indipendentemente dall'istanza di contenitore su cui vengono create tali attività.
4. Crea un servizio Amazon ECS con tre istanze della definizione dell'attività.

Stack tecnologico

- Amazon EC2
- Amazon ECS
- Amazon EFS

Tools (Strumenti)

- Amazon EC2 — Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi usare Amazon EC2 per lanciare tutti o pochi server virtuali di cui hai bisogno, con scalabilità orizzontale o orizzontale.
- Amazon ECS — Amazon Elastic Container Service (Amazon ECS) è un servizio di gestione dei container veloce e altamente scalabile per l'esecuzione, l'arresto e la gestione dei container su un cluster. Puoi eseguire le tue attività e i tuoi servizi su un'infrastruttura serverless gestita da AWS

Fargate. In alternativa, per un maggiore controllo sulla tua infrastruttura, puoi eseguire le tue attività e i tuoi servizi su un cluster di EC2 istanze da te gestito.

- Amazon EFS — Amazon Elastic File System (Amazon EFS) fornisce un file system NFS elastico semplice, scalabile e completamente gestito da utilizzare con i servizi cloud AWS e le risorse locali.
- AWS CLI — L'AWS Command Line Interface (AWS CLI) è uno strumento open source per interagire con i servizi AWS tramite comandi nella shell della riga di comando. Con una configurazione minima, puoi eseguire comandi AWS CLI che implementano funzionalità equivalenti a quelle fornite dalla Console di gestione AWS basata su browser da un prompt dei comandi.

Epiche

Creare un file system Amazon EFS

Operazione	Description	Competenze richieste
Crea un file system Amazon EFS utilizzando la Console di gestione AWS.	1. **Note** Crea un file system Amazon EFS e scegli il VPC che include i tuoi contenitori. : se utilizzi un VPC diverso, configura una connessione peering VPC. 2. Prendi nota dell'ID file system.	AWS DevOps

Crea una definizione di attività Amazon ECS utilizzando un file system Amazon EFS o l'AWS CLI

Operazione	Description	Competenze richieste
Crea una definizione di attività utilizzando un file system Amazon EFS.	Crea una definizione di attività utilizzando la nuova console Amazon ECS o la classica console Amazon ECS con le seguenti configurazioni: • Se utilizzi la nuova console, scegli Amazon EC2 instances for App environment. Se usi la console classica, scegli EC2come tipo di avvio. • Aggiungi un volume. Immettete un nome per il volume, scegliete EFS per il tipo di volume, quindi scegliete l'ID del file system che avete annotato in precedenza. Per la directory principale, scegli il percorso del file system Amazon EFS che desideri ospitare sull'host del contenitore Amazon ECS.	AWS DevOps
Crea una definizione di attività utilizzando la CLI di AWS.	1. Per creare un modello JSON con segnapost o dei parametri di input per la definizione dell'attività, esegui il seguente comando:	AWS DevOps

Operazione	Description	Competenze richieste

	`aws ecs register-task-definition --generate-cli-skeleton`
	2. Per creare la definizione dell'attività con il modello JSON, esegui il comando seguente:
	`aws ecs register-task-definition --cli-input-json file://<path_to_your_json_file>`
	3. **Note**
	Immettete i parametri di input nel modello JSON in base al `task_definition_parameters.json` file (allegato). : per ulteriori informazioni sui parametri di input, consulta Parametri di definizione delle attività (documentazione di Amazon ECS) e register-task-definition(AWS	

Operazione	Description	Competenze richieste
	CLI Command Reference).	

Risorse correlate

- Definizioni delle attività di Amazon ECS
- Volumi Amazon EFS

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Implementa le funzioni Lambda con immagini dei container

Ram Kandaswamy, Amazon Web Services

Riepilogo

AWS Lambda supporta le immagini dei contenitori come modello di distribuzione. Questo modello mostra come distribuire le funzioni Lambda tramite immagini dei contenitori.

Lambda è un servizio di elaborazione serverless e basato sugli eventi che puoi utilizzare per eseguire codice praticamente per qualsiasi tipo di applicazione o servizio di backend senza dover fornire o gestire server. Con il supporto di immagini container per le funzioni Lambda, ottieni i vantaggi di un massimo di 10 GB di spazio di archiviazione per gli elementi dell'applicazione e la possibilità di utilizzare strumenti familiari per lo sviluppo di immagini container.

L'esempio in questo modello utilizza Python come linguaggio di programmazione sottostante, ma è possibile utilizzare altri linguaggi, come Java, Node.js o Go. Per la fonte, prendi in considerazione un sistema basato su Git come GitHub, o Bitbucket GitLab, oppure usa Amazon Simple Storage Service (Amazon S3).

Prerequisiti e limitazioni

Prerequisiti

- Amazon Elastic Container Registry (Amazon ECR) attivato
- Codice dell'applicazione
- Immagini Docker con il client di interfaccia runtime e l'ultima versione di Python
- Conoscenza pratica di Git

Limitazioni

- La dimensione massima dell'immagine supportata è di 10 GB.
- L'autonomia massima per una distribuzione di container basata su Lambda è di 15 minuti.

Architecture

Architettura di Target

1. Si crea un repository Git e si esegue il commit del codice dell'applicazione nel repository.
2. Il AWS CodeBuild progetto viene attivato dal commit delle modifiche.
3. Il CodeBuild progetto crea l'immagine Docker e pubblica l'immagine creata su Amazon ECR.
4. La funzione Lambda viene creata utilizzando l'immagine in Amazon ECR.

Automazione e scalabilità

Questo modello può essere automatizzato utilizzando AWS CloudFormation o operazioni API da un SDK. AWS Cloud Development Kit (AWS CDK) Lambda può scalare automaticamente in base al numero di richieste e puoi ottimizzarlo utilizzando i parametri di concorrenza. Per ulteriori informazioni, consulta la documentazione di Lambda.

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationAWS CloudFormationhelps configuri AWS le risorse, le fornisci in modo rapido e coerente e le gestisci per tutto il loro ciclo di vita su Account AWS e. Regioni AWS Questo modello utilizza AWS CloudFormation Application Composer, che consente di visualizzare e modificare visivamente i modelli. CloudFormation
- AWS CodeBuildè un servizio di compilazione completamente gestito che consente di compilare il codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.

Altri strumenti

- Docker è un insieme di prodotti Platform as a Service (PaaS) che utilizzano la virtualizzazione a livello di sistema operativo per fornire software in container.
- GitHubGitLab, e Bitbucket sono alcuni dei sistemi di controllo del codice sorgente basati su Git comunemente usati per tenere traccia delle modifiche al codice sorgente.

Best practice

- Rendi la tua funzione il più efficiente e ridotta possibile per evitare di caricare file non necessari.
- Cerca di avere livelli statici più in alto nell'elenco dei file Docker e posiziona i livelli che cambiano più spesso più in basso. Ciò migliora la memorizzazione nella cache, che migliora le prestazioni.
- Il proprietario dell'immagine è responsabile dell'aggiornamento e della correzione dell'immagine. Aggiungete questa cadenza di aggiornamento ai vostri processi operativi. Per ulteriori informazioni, consulta la documentazione relativa ad AWS Lambda.

Epiche

Crea un progetto in CodeBuild

Operazione	Description	Competenze richieste
Crea un repository Git.	Crea un repository Git che conterrà il codice sorgente dell'applicazione, il Dockerfile e il file. `buildspec.yaml`	Developer
Crea un progetto. CodeBuild	Per utilizzare un CodeBuild progetto per creare l'immagine Lambda personalizzata, procedi come segue: 1. Accedi a e apri Console di gestione AWS la CodeBuild console all'indirizzo https://console.aws.amazon.com/codesuite/ codebuild/. 2. Crea un nuovo progetto. Come sorgente, scegli il repository Git che hai creato. Per informazioni sui diversi tipi di integrazione del repository Git, consulta	Developer

Operazione	Description	Competenze richieste
	la documentazione Working with connections. 3. Conferma che la modalità privilegiata sia abilitata . Per creare immagini Docker, questo è necessario. Altrimenti, l'immagine non verrà creata correttamente. 4. Fornite i valori per il nome e la descrizione del progetto.	
Modifica il Dockerfile.	Il Dockerfile dovrebbe trovarsi nella directory di primo livello in cui stai sviluppando l'applicazione. Il codice Python dovrebbe trovarsi nella cartella`src`. Quando crei l'immagine, usa le immagini ufficiali supportate da Lambda. In caso contrario, si verificherà un errore di bootstrap che renderà più difficile il processo di compressione. Per i dettagli, consulta la sezione Informazioni aggiuntive.	Developer

Operazione	Description	Competenze richieste
Crea un repository in Amazon ECR.	Crea un repository di contenitori in Amazon ECR. Nel seguente comando di esempio, il nome del repository creato è: `cf-demo` `aws ecr create-repository --cf-demo` Il repository verrà referenziato nel file. `buildspec.yaml`	Amministratore AWS, sviluppatore
Invia l'immagine ad Amazon ECR.	È possibile utilizzare CodeBuild per eseguire il processo di creazione dell'immagine. CodeBuild necessita dell'autorizzazione per interagire con Amazon ECR e lavorare con S3. Come parte del processo, l'immagine Docker viene creata e inserita nel registro Amazon ECR. Per i dettagli sul modello e sul codice, consulta la sezione Informazioni aggiuntive.	Developer

Operazione	Description	Competenze richieste
Verifica che l'immagine sia nel repository.	Per verificare che l'immagine sia presente nel repository, sulla console Amazon ECR, scegli Repositories. L'immagine dovrebbe essere elencata, con i tag e con i risultati di un rapporto di scansione delle vulnerabilità se tale funzionalità è stata attivata nelle impostazioni di Amazon ECR. Per ulteriori informazioni, consulta la documentazione di AWS.	Developer

Crea la funzione Lambda per eseguire l'immagine

Operazione	Description	Competenze richieste
Creazione della funzione Lambda	Sulla console Lambda, scegli Crea funzione, quindi scegli Immagine contenitore. Inserisci il nome della funzione e l'URI per l'immagine che si trova nel repository Amazon ECR, quindi scegli Crea funzione. Per ulteriori informazioni, consulta la documentazione di AWS Lambda.	Sviluppatore di app
Prova la funzione Lambda.	Per richiamare e testare la funzione, scegli Test. Per ulteriori informazioni, consulta	Sviluppatore di app

Operazione	Description	Competenze richieste
	la documentazione di AWS Lambda.	

risoluzione dei problemi

Problema	Soluzione
La compilazione non ha esito positivo.	1. Controlla se la modalità privilegiata è attiva per il CodeBuild progetto. 2. Assicurati che i comandi relativi a Docker abbiano le autorizzazioni necessarie. Sto provando ad aggiungere `sudo` ai comandi. 3. Verifica che il ruolo IAM associato a CodeBuild abbia una policy con azioni appropriate per interagire con Amazon ECR, Amazon S3 e i log. CloudWatch

Risorse correlate

- Immagini di base per Lambda
- Esempio di Docker per CodeBuild
- Passa credenziali temporanee

Informazioni aggiuntive

Modifica il Dockerfile

Il codice seguente mostra i comandi che modifichi nel Dockerfile:

```
FROM public.ecr.aws/lambda/python:3.xx

# Copy function code
COPY app.py ${LAMBDA_TASK_ROOT}
COPY requirements.txt  ${LAMBDA_TASK_ROOT}
```

```
# install dependencies
RUN pip3 install --user -r requirements.txt

# Set the CMD to your handler (could also be done as a parameter override outside of
 the Dockerfile)
CMD [ "app.lambda_handler" ]
```

Nel FROM comando, usa il valore appropriato per la versione di Python supportata da Lambda (ad esempio,). `3.12` Questa sarà l'immagine di base disponibile nell'archivio pubblico di immagini Amazon ECR.

Il `COPY app.py ${LAMBDA_TASK_ROOT}` comando copia il codice nella directory principale dell'attività, che verrà utilizzata dalla funzione Lambda. Questo comando utilizza la variabile di ambiente, quindi non dobbiamo preoccuparci del percorso effettivo. La funzione da eseguire viene passata come argomento al `CMD [ "app.lambda_handler" ]` comando.

Il `COPY requirements.txt` comando acquisisce le dipendenze necessarie per il codice.

Il `RUN pip install --user -r requirements.txt` comando installa le dipendenze nella directory utente locale.

Per creare l'immagine, esegui il comando seguente.

```
docker build -t <image name> .
```

Aggiungi l'immagine in Amazon ECR

Nel codice seguente, sostituiscilo `aws_account_id` con il numero di account e sostituiscilo `us-east-1` se utilizzi una regione diversa. Il `buildspec` file utilizza il numero di CodeBuild build per identificare in modo univoco le versioni delle immagini come valore del tag. Puoi modificarlo in base alle tue esigenze.

Il codice personalizzato buildspec

```
phases:
  install:
    runtime-versions:
       python: 3.xx
  pre_build:
    commands:
```

```
      - python3 --version
      - pip3 install --upgrade pip
      - pip3 install --upgrade awscli
      - sudo docker info
  build:
    commands:
      - echo Build started on `date`
      - echo Building the Docker image...
      - ls
      - cd app
      - docker build -t cf-demo:$CODEBUILD_BUILD_NUMBER .
      - docker container ls
  post_build:
    commands:
      - echo Build completed on `date`
      - echo Pushing the Docker image...
      - aws ecr get-login-password --region us-east-1 | docker login --username AWS --
password-stdin aws_account_id.dkr.ecr.us-east-1.amazonaws.com
      - docker tag cf-demo:$CODEBUILD_BUILD_NUMBER aws_account_id.dkr.ecr.us-
east-1.amazonaws.com/cf-demo:$CODEBUILD_BUILD_NUMBER
      - docker push aws_account_id.dkr.ecr.us-east-1.amazonaws.com/cf-demo:
$CODEBUILD_BUILD_NUMBER
```

Distribuisci microservizi Java su Amazon ECS utilizzando AWS Fargate

Vijay Thompson e Sandeep Bondugula, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni per la distribuzione di microservizi Java containerizzati su Amazon Elastic Container Service (Amazon ECS) utilizzando AWS Fargate. Il modello non utilizza Amazon Elastic Container Registry (Amazon ECR) per la gestione dei container; le immagini Docker vengono invece estratte da un hub Docker.

Prerequisiti e limitazioni

Prerequisiti

- Un'applicazione di microservizi Java esistente su un hub Docker
- Un repository Docker pubblico
- Un account AWS attivo
- Familiarità con i servizi AWS, tra cui Amazon ECS e Fargate
- Framework Docker, Java e Spring Boot
- Amazon Relational Database Service (Amazon RDS) attivo e funzionante (opzionale)
- Un cloud privato virtuale (VPC) se l'applicazione richiede Amazon RDS (opzionale)

Architecture

Stack tecnologico di origine

- Microservizi Java (ad esempio, implementati in Spring Boot) e distribuiti su Docker

Architettura di origine

Stack tecnologico Target

- Un cluster Amazon ECS che ospita ogni microservizio utilizzando Fargate
- Una rete VPC per ospitare il cluster Amazon ECS e i gruppi di sicurezza associati
- Una cluster/task definizione per ogni microservizio che attiva i contenitori utilizzando Fargate

Architettura Target

Tools (Strumenti)

Strumenti

- Amazon ECS elimina la necessità di installare e utilizzare il proprio software di orchestrazione dei container, gestire e scalare un cluster di macchine virtuali o pianificare contenitori su tali macchine virtuali.
- AWS Fargate ti aiuta a eseguire container senza dover gestire server o istanze Amazon Elastic Compute Cloud (Amazon). EC2 Viene utilizzato insieme ad Amazon Elastic Container Service (Amazon ECS).
- Docker è una piattaforma software che consente di creare, testare e distribuire applicazioni rapidamente. Docker impacchetta il software in unità standardizzate chiamate contenitori che contengono tutto ciò di cui il software ha bisogno per funzionare, tra cui librerie, strumenti di sistema, codice e runtime.

Codice Docker

Il seguente Dockerfile specifica la versione di Java Development Kit (JDK) utilizzata, in cui esiste il file di archivio Java (JAR), il numero di porta esposto e il punto di ingresso per l'applicazione.

```
FROM openjdk:11
ADD target/Spring-docker.jar Spring-docker.jar
EXPOSE 8080
ENTRYPOINT ["java","-jar","Spring-docker.jar"]
```

Epiche

Crea nuove definizioni di attività

Operazione	Description	Competenze richieste
Crea una definizione di attività.	L'esecuzione di un contenitore Docker in Amazon ECS richiede una definizione di attività. Apri la console	Amministratore di sistema AWS, sviluppatore di app

Operazione	Description	Competenze richieste
	Amazon ECS all'indirizzo https://console.aws.amazon.com/ecs/, scegli Definizioni attività, quindi crea una nuova definizione di attività. Per ulteriori informazioni, consulta la documentazione di Amazon ECS.	
Scegli il tipo di lancio.	Scegli Fargate come tipo di lancio.	Amministratore di sistema AWS, sviluppatore di app
Configura l'attività.	Definire un nome per l'attività e configurare l'applicazione con la quantità appropriata di memoria e CPU.	Amministratore di sistema AWS, sviluppatore di app
Definisci il contenitore.	Specificate il nome del contenitore. Per l'immagine, inserisci il nome del sito Docker, il nome del repository e il nome del tag dell'immagine Docker (). `docker.io/sample-repo/sample-application:sample-tag-name` Imposta i limiti di memoria per l'applicazione e imposta le mappature delle porte (`8080, 80`) per le porte consentite.	Amministratore di sistema AWS, sviluppatore di app

Operazione	Description	Competenze richieste
Crea l'attività.	Quando le configurazioni dell'attività e del contenito re sono a posto, crea l'attivit à. Per istruzioni dettagliate, consulta i collegamenti nella sezione Risorse correlate.	Amministratore di sistema AWS, sviluppatore di app

Configurazione del cluster

Operazione	Description	Competenze richieste
Crea e configura un cluster.	Scegli Solo rete come tipo di cluster, configura il nome, quindi crea il cluster o utilizza un cluster esistente, se disponibile. Per ulteriori informazioni, consulta la documentazione di Amazon ECS.	Amministratore di sistema AWS, sviluppatore di app

Configura Task

Operazione	Description	Competenze richieste
Crea un’attività .	All'interno del cluster, scegli Esegui nuova attività.	Amministratore di sistema AWS, sviluppatore di app
Scegli il tipo di lancio.	Scegli Fargate come tipo di lancio.	Amministratore di sistema AWS, sviluppatore di app
Scegli la definizione dell'atti vità, la revisione e la versione della piattaforma.	Scegli l'attività che desideri eseguire, la revisione della definizione dell'attività e la versione della piattaforma.	Amministratore di sistema AWS, sviluppatore di app

Operazione	Description	Competenze richieste
Seleziona il cluster .	Scegli il cluster da cui desideri eseguire l'attività.	Amministratore di sistema AWS, sviluppatore di app
Specificare il numero di attività.	Configura il numero di attività da eseguire. Se si avvia con due o più attività, è necessari o un sistema di bilanciamento del carico per distribuire il traffico tra le attività.	Amministratore di sistema AWS, sviluppatore di app
Specificare il gruppo di attività.	(Facoltativo) Specificate il nome di un gruppo di attività per identificare un insieme di attività correlate come gruppo di attività.	Amministratore di sistema AWS, sviluppatore di app
Configura il VPC del cluster, le sottoreti e i gruppi di sicurezza .	Configura il VPC del cluster e le sottoreti su cui desideri distribuire l'applicazione. Crea o aggiorna gruppi di sicurezza (HTTP, HTTPS e porta 8080) per fornire l'accesso alle connessioni in entrata e in uscita.	Amministratore di sistema AWS, sviluppatore di app
Configura le impostazioni IP pubbliche.	Abilita o disabilita l'IP pubblico, a seconda che desideri utilizzare un indirizzo IP pubblico per le attività di Fargate. L'opzione predefinita e consigliata è Abilitata.	Amministratore di sistema AWS, sviluppatore di app
Rivedi le impostazioni e crea l'attività	Controlla le impostazioni, quindi scegli Esegui operazion e.	Amministratore di sistema AWS, sviluppatore di app

Tagliare

Operazione	Description	Competenze richieste
Copia l'URL dell'applicazione.	Quando lo stato dell'attività è stato aggiornato a In esecuzione, seleziona l'attività. Nella sezione Rete, copia l'IP pubblico.	Amministratore di sistema AWS, sviluppatore di app
Testa la tua applicazione.	Nel browser, inserisci l'IP pubblico per testare l'applicazione.	Amministratore di sistema AWS, sviluppatore di app

Risorse correlate

- Nozioni di base su Docker per Amazon ECS (documentazione Amazon ECS)
- Amazon ECS su AWS Fargate (documentazione Amazon ECS)
- Creazione di una definizione di attività (documentazione Amazon ECS)
- Creazione di un cluster (documentazione Amazon ECS)
- Configurazione dei parametri di base del servizio (documentazione Amazon ECS)
- Configurazione di una rete (documentazione Amazon ECS)
- Implementazione di microservizi Java su Amazon ECS (post di blog)

Distribuisci risorse e pacchetti Kubernetes utilizzando Amazon EKS e un repository di grafici Helm in Amazon S3

Sagar Panigrahi, Amazon Web Services

Riepilogo

Questo modello ti aiuta a gestire le applicazioni Kubernetes in modo efficiente, indipendentemente dalla loro complessità. Il modello integra Helm nelle pipeline esistenti di integrazione continua e distribuzione continua (CI/CD) per distribuire le applicazioni in un cluster Kubernetes. Helm è un gestore di pacchetti Kubernetes che ti aiuta a gestire le applicazioni Kubernetes. I grafici Helm aiutano a definire, installare e aggiornare applicazioni Kubernetes complesse. I grafici possono essere modificati in versioni e archiviati negli archivi Helm, il che migliora il tempo medio di ripristino (MTTR) durante le interruzioni.

Questo modello utilizza Amazon Elastic Kubernetes Service (Amazon EKS) per il cluster Kubernetes. Utilizza Amazon Simple Storage Service (Amazon S3) come repository di grafici Helm, in modo che i grafici possano essere gestiti centralmente e accessibili dagli sviluppatori di tutta l'organizzazione.

Prerequisiti e limitazioni

Prerequisiti

- Un account Amazon Web Services (AWS) attivo con un cloud privato virtuale (VPC)
- Cluster Amazon EKS
- Nodi di lavoro configurati all'interno del cluster Amazon EKS e pronti a gestire carichi di lavoro
- Kubectl per configurare il file Amazon EKS kubeconfig per il cluster di destinazione nel computer client
- Accesso ad AWS Identity and Access Management (IAM) per creare il bucket S3
- Accesso IAM (programmatico o di ruolo) ad Amazon S3 dal computer client
- Gestione del codice sorgente e pipeline CI/CD

Limitazioni

- Al momento non è disponibile alcun supporto per l'aggiornamento, l'eliminazione o la gestione delle definizioni di risorse personalizzate (). CRDs

- Se si utilizza una risorsa che fa riferimento a un CRD, il CRD deve essere installato separatamente (al di fuori del grafico).

Versioni del prodotto

- Helm v3.6.3

Architecture

Stack tecnologico Target

- Amazon EKS
- Amazon VPC
- Simple Storage Service (Amazon S3)
- Gestione del codice sorgente
- Helm
- Kubectl

Architettura di destinazione

Automazione e scalabilità

- AWS CloudFormation può essere utilizzato per automatizzare la creazione dell'infrastruttura. Per ulteriori informazioni, consulta Creazione di risorse Amazon EKS con AWS CloudFormation nella documentazione di Amazon EKS.
- Helm deve essere incorporato nel tuo strumento di CI/CD automazione esistente per automatizzare il packaging e il controllo delle versioni dei grafici Helm (al di fuori dell'ambito di questo schema).
- GitVersion oppure è possibile utilizzare i numeri di build Jenkins per automatizzare il controllo delle versioni dei grafici.

Tools (Strumenti)

Strumenti

- Amazon EKS — Amazon Elastic Kubernetes Service (Amazon EKS) è un servizio gestito per eseguire Kubernetes su AWS senza dover installare o mantenere il proprio piano di controllo Kubernetes. Kubernetes è un sistema open source per automatizzare l'implementazione, il dimensionamento e la gestione di applicazioni containerizzate.
- Helm — Helm è un gestore di pacchetti per Kubernetes che ti aiuta a installare e gestire applicazioni sul tuo cluster Kubernetes.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è uno storage per Internet. È possibile utilizzare Amazon S3 per memorizzare e recuperare qualsiasi volume di dati, in qualunque momento e da qualunque luogo tramite il Web.
- Kubectl — Kubectl è un'utilità da riga di comando per l'esecuzione di comandi su cluster Kubernetes.

Codice

Il codice di esempio è allegato.

Epiche

Configura e inizializza Helm

Operazione	Description	Competenze richieste
Installa il client Helm.	Per scaricare e installare il client Helm sul sistema locale, utilizzare il seguente comando. `sudo curl https://raw.githubusercontent.com/helm/helm/master/scripts/get-helm-3 | bash`	DevOps ingegnere
Convalida l'installazione di Helm.	Per verificare che Helm sia in grado di comunicare con il server API Kubernetes all'inter	DevOps ingegnere

Operazione	Description	Competenze richieste
	no del cluster Amazon EKS, esegui. `helm version`	

Crea e installa un grafico Helm nel cluster Amazon EKS

Operazione	Description	Competenze richieste
Crea un grafico Helm per NGINX.	Per creare un diagramma Helm denominato `my-nginx` sul computer client, esegui. `helm create my-nginx`	DevOps ingegnere
Esamina la struttura del grafico.	Per rivedere la struttura del grafico, esegui il comando `tree` `tree my-nginx/` .	DevOps ingegnere
Disattiva la creazione dell'account di servizio nel grafico.	Nella `serviceAccount` sezione `values.yaml` , imposta la `create` chiave su. `false` Questa opzione è disattivata perché non è necessario creare un account di servizio per questo pattern.	DevOps ingegnere
Convalida (lint) il grafico modificato per eventuali errori sintattici.	Per convalidare il grafico per eventuali errori sintattici prima di installarlo nel cluster di destinazione, esegui. `helm lint my-nginx/`	DevOps ingegnere
Installa il grafico per distribuire le risorse Kubernetes.	Per eseguire l'installazione di Helm chart, usa il comando seguente. `helm install --name my-nginx-release --debug`	DevOps ingegnere

Operazione	Description	Competenze richieste
	`my-nginx/ --namespace helm-space` Il `debug` flag opzionale emette tutti i messaggi di debug durante l'installazione. Il `namespace` flag specifica lo spazio dei nomi in cui verranno create le risorse che fanno parte di questo grafico.	
Esamina le risorse nel cluster Amazon EKS.	Per esaminare le risorse che sono state create come parte del grafico Helm nel `helm-space` namespace, usa il comando seguente. `kubectl get all -n helm-space`	DevOps ingegnere

Torna a una versione precedente di un'applicazione Kubernetes

Operazione	Description	Competenze richieste
Modifica e aggiorna la versione.	Per modificare il grafico, in`values.yaml` , modifica il `replicaCount` valore in2. Quindi aggiorna la versione già installata eseguendo il comando seguente. `helm upgrade my-nginx-release my-nginx/ --namespace helm-space`	DevOps ingegnere

Operazione	Description	Competenze richieste
Consulta la cronologia della versione di Helm.	Per elencare tutte le revisioni di una versione specifica che sono state installate utilizzan do Helm, esegui il comando seguente. `helm history my-nginx-release`	DevOps ingegnere
Esamina i dettagli per una revisione specifica.	Prima di passare o ripristin are una versione funzionan te e per un ulteriore livello di convalida prima di installar e una revisione, visualizza quali valori sono stati passati a ciascuna delle revisioni utilizzando il comando seguente. `helm get --revision=2 my-nginx-release`	DevOps ingegnere
Torna a una versione precedente.	Per tornare a una revisione precedente, usa il seguente comando. `helm rollback my-nginx-release 1` In questo esempio viene eseguito il rollback alla revisione numero 1.	DevOps ingegnere

Inizializza un bucket S3 come repository Helm

Operazione	Description	Competenze richieste
Crea un bucket S3 per i grafici Helm.	Crea un bucket S3 unico. Nel bucket, crea una cartella chiamata. `charts` L'esempio di questo modello utilizza `s3://my-helm-charts/charts` come archivio grafico di destinazione.	Amministratore cloud
Installa il plug-in Helm per Amazon S3.	Per installare il plugin helm-s3 sul tuo computer client, usa il seguente comando. `helm plugin install https://github.com/hypnoglow/helm-s3.git --version 0.10.0` Nota: il supporto per Helm V3 è disponibile con la versione del plugin 0.9.0 e successive.	DevOps ingegnere
Inizializza il repository Amazon S3 Helm.	Per inizializzare la cartella di destinazione come repository Helm, usa il seguente comando. `helm S3 init s3://my-helm-charts/charts` Il comando crea un `index.yaml` file nella destinazione per tenere traccia di tutte le informazioni del	DevOps ingegnere

Operazione	Description	Competenze richieste
	grafico archiviate in quella posizione.	
Aggiungi il repository Amazon S3 a Helm.	Per aggiungere il repository nel computer client, usa il seguente comando. `helm repo add my-helm-charts s3://my-helm-charts/charts` Questo comando aggiunge un alias al repository di destinazione nel computer client Helm.	DevOps ingegnere
Controlla l'elenco dei repository.	Per visualizzare l'elenco dei repository nel computer client Helm, esegui. `helm repo list`	DevOps ingegnere

Package e memorizza i grafici nel repository Amazon S3 Helm

Operazione	Description	Competenze richieste
Creazione pacchetto del grafico.	Per impacchettare il `my-nginx` grafico che hai creato, eseguihelm `package ./my-nginx/` . Il comando racchiude tutto il contenuto della cartella del `my-nginx` grafico in un file di archivio, denominato utilizzando il numero di versione indicato nel `Chart.yaml` file.	DevOps ingegnere

Operazione	Description	Competenze richieste
Archivia il pacchetto nel repository Amazon S3 Helm.	Per caricare il pacchetto nel repository Helm in Amazon S3, esegui il comando seguente, utilizzando il nome corretto del file. `.tgz` `helm s3 push ./my-ngin x-0.1.0.tgz my-helm-c harts`	DevOps ingegnere
Cerca la carta Helm.	Per confermare che il grafico sia visualizzato sia localment e che nel repository Helm in Amazon S3, esegui il comando seguente. `helm search repo my- nginx`	DevOps ingegnere

Modifica, versione e impacchetta un grafico

Operazione	Description	Competenze richieste
Modifica e impacchetta il grafico.	In`values.yaml` , imposta il `replicaCount` valore su`1`. Quindi impacchetta il grafico eseguendolo`helm package ./my-nginx/` , questa volta cambiando la versione `Chart.yaml` in`0.1.1`. Il controllo delle versioni viene idealmente aggiornat	DevOps ingegnere

Operazione	Description	Competenze richieste
	o tramite l'automazione utilizzando strumenti come Jenkins GitVersion o Jenkins build Numbers in una CI/CD pipeline. L'automazione del numero di versione non rientra nell'ambito di questo schema.	
Invia la nuova versione al repository Helm in Amazon S3.	Per inviare il nuovo pacchetto con la versione 0.1.1 al repository `my-helm-charts` Helm in Amazon S3, esegui il comando seguente. `helm s3 push ./my-nginx-0.1.1.tgz my-helm-charts`	DevOps ingegnere

Cerca e installa un grafico dal repository Amazon S3 Helm

Operazione	Description	Competenze richieste
Cerca tutte le versioni del grafico my-nginx.	Per visualizzare tutte le versioni disponibili di un grafico, esegui il seguente comando con il flag. `--versions` `helm search repo my-nginx --versions` Senza il flag, per impostazi one predefinita Helm visualizz	DevOps ingegnere

Operazione	Description	Competenze richieste
	a l'ultima versione caricata di un grafico.	
Installa un grafico dal repository Amazon S3 Helm.	I risultati della ricerca dell'attività precedente mostrano le diverse versioni del grafico. `my-nginx` Per installare la nuova versione (0.1.1) dal repository Amazon S3 Helm, usa il seguente comando. `helm upgrade my-nginx-release my-helm-charts/my-nginx --version 0.1.1 --namespace helm-space`	DevOps ingegnere

Risorse correlate

- documentazione HELM
- plugin helm-s3 (licenza MIT)
- File binario del client HELM
- Documentazione Amazon EKS

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Implementa un cluster CockroachDB in Amazon EKS utilizzando Terraform

Sandip Gangapadhyay e Kalyan Senthilnathan, Amazon Web Services

Riepilogo

Questo modello fornisce un modulo HashiCorp Terraform per la distribuzione di un cluster CockroachDB multinodo su Amazon Elastic Kubernetes Service (Amazon EKS) utilizzando l'operatore CockroachDB. CockroachDB è un database SQL distribuito che fornisce sharding orizzontale automatico, alta disponibilità e prestazioni coerenti in cluster distribuiti geograficamente. Questo modello utilizza Amazon EKS come piattaforma Kubernetes gestita e implementa cert-manager per comunicazioni tra nodi protette da TLS. Utilizza anche un Network Load Balancer per la distribuzione del traffico e crea CockroachDB StatefulSetscon pod che replicano automaticamente i dati per la tolleranza ai guasti e le prestazioni.

Destinatari

Per implementare questo modello, ti consigliamo di conoscere quanto segue:

- HashiCorp Concetti e pratiche di Terraform Infrastructure as Code (IaC)
- Servizi AWS, in particolare Amazon EKS
- Nozioni di base su Kubernetes, tra cui operatori e configurazioni dei StatefulSets servizi
- Database SQL distribuiti
- Concetti di sicurezza, come la gestione dei certificati TLS.
- DevOps pratiche, CI/CD flussi di lavoro e automazione dell'infrastruttura

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Autorizzazioni per distribuire risorse in un cluster Amazon EKS
- Un cluster Amazon EKS versione 1.23 o successiva, con nodi etichettati `node=cockroachdb`
- Driver Amazon Elastic Block Store Container Storage Interface (CSI) versione 1.19.0 o successiva, installato nel cluster Amazon EKS
- Terraform CLI versione 1.0.0 o successiva, installata

- kubectl, installato
- Git, installato
- AWS Command Line Interface (AWS CLI) versione 2.9.18 o successiva, installata e configurata

Limitazioni

- L'operatore CockroachDB Kubernetes non supporta più cluster Kubernetes per implementazioni multiregionali. Per ulteriori limitazioni, consulta Orchestrate CockroachDB su più cluster Kubernetes (documentazione CockroachDB) e CockroachDB Kubernetes Operator (). GitHub
- L'eliminazione automatica delle dichiarazioni di PVCs volume persistenti () è attualmente disabilitata per impostazione predefinita. Ciò significa che dopo la disattivazione e la rimozione di un nodo, l'operatore non rimuoverà il volume persistente che era montato sul relativo pod. Per ulteriori informazioni, consulta Potatura automatica del PVC nella documentazione di CockroachDB.

Versioni del prodotto

- CockroachDB versione 22.2.2

Architecture

Architettura Target

Il diagramma seguente mostra un'implementazione di CockroachDB ad alta disponibilità in tre zone di AWS disponibilità all'interno di un cloud privato virtuale (VPC). I pod CockroachDB sono gestiti tramite Amazon EKS. L'architettura illustra come gli utenti accedono al database tramite un Network Load Balancer, che distribuisce il traffico ai pod CockroachDB. I pod vengono eseguiti su istanze Amazon Elastic Compute Cloud EC2 (Amazon) in ogni zona di disponibilità, il che offre resilienza e tolleranza agli errori.

Risorse create

L'implementazione del modulo Terraform utilizzato in questo modello crea le seguenti risorse:

1. Network Load Balancer: questa risorsa funge da punto di ingresso per le richieste dei client e distribuisce uniformemente il traffico tra le istanze CockroachDB.

2. CockroachDB StatefulSet: StatefulSet definisce lo stato desiderato della distribuzione di CockroachDB all'interno del cluster Amazon EKS. Gestisce la distribuzione ordinata, il ridimensionamento e gli aggiornamenti dei pod CockroachDB.
3. Pods CockroachDB: questi pod sono istanze di CockroachDB in esecuzione come contenitori all'interno dei pod Kubernetes. Questi pod archiviano e gestiscono i dati in tutto il cluster distribuito.
4. Database CockroachDB: questo è il database distribuito gestito da CockroachDB, che si estende su più pod. Replica i dati per garantire disponibilità, tolleranza agli errori e prestazioni elevate.

Tools (Strumenti)

Servizi AWS

- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando.
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire AWS Kubernetes senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.

Altri strumenti

- HashiCorp Terraform è uno strumento Infrastructure as Code (IaC) che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud.
- kubectl è un'interfaccia a riga di comando che ti aiuta a eseguire comandi sui cluster Kubernetes.

Archivio di codice

Il codice per questo pattern è disponibile nel cluster GitHub Deploy a CockroachDB in Amazon EKS utilizzando il repository Terraform. L'archivio di codice contiene i seguenti file e cartelle per Terraform:

- `modules`folder — Questa cartella contiene il modulo Terraform per CockroachDB
- `main`folder: questa cartella contiene il modulo root che chiama il modulo secondario CockroachDB per creare il cluster di database CockroachDB.

Best practice

- Non ridimensionare a meno di tre nodi. Questo è considerato un anti-pattern su CockroachDB e può causare errori. Per ulteriori informazioni, consulta Cluster Scaling nella documentazione di CockroachDB.
- Implementa la scalabilità automatica di Amazon EKS utilizzando Karpernter o Cluster Autoscaler. Ciò consente al cluster CockroachDB di scalare orizzontalmente e di aggiungere nuovi nodi automaticamente. Per ulteriori informazioni, consulta Scale cluster compute with Karpenter e Cluster Autoscaler nella documentazione di Amazon EKS.

> **Note**
>
> Grazie alla regola di pianificazione `podAntiAffinity` Kubernetes, è possibile pianificare un solo pod CockroachDB in un nodo Amazon EKS.

- Per le best practice di sicurezza di Amazon EKS, consulta Best Practices for Security nella documentazione di Amazon EKS.
- Per le best practice relative alle prestazioni SQL per CockroachDB, consulta SQL Performance Best Practices nella documentazione di CockroachDB.
- Per ulteriori informazioni sulla configurazione di un backend remoto Amazon Simple Storage Service (Amazon S3) per il file di stato Terraform, consulta Amazon S3 nella documentazione di Terraform.

Epiche

Configurare l'ambiente

Operazione	Description	Competenze richieste
Clona il repository del codice.	Inserisci il seguente comando per clonare il repository: `git clone https://github.com/aws-samples/crdb-cluster-eks-terraform.git`	DevOps ingegnere, Git

Operazione	Description	Competenze richieste
Aggiorna le variabili Terraform.	1. Inserisci il seguente comando per navigare nella cartella principale del repository clonato: `cd crdb-cluster-eks-terraform/main` 2. Apri il file variable.tf. 3. Configura il valore predefinito per le seguenti variabili: • `region`— Inserire l'obiettivo Regione AWS • `eks_cluster_name` — Inserisci il nome del cluster Amazon EKS di destinazione • `number_of_nodes` — Inserisci il numero di nodi da implementare 4. Salvate e chiudete il file Variable.tf.	DevOps ingegnere, Terraform

Implementa le risorse

Operazione	Description	Competenze richieste
Implementa l'infrastruttura.	1. Inserisci il seguente comando per inizializzare la distribuzione di Terraform: `terraform init`	DevOps ingegnere, Terraform

Operazione	Description	Competenze richieste

	2. Immettere il seguente comando per generare un piano di esecuzione:
	`terraform plan`
	3. Rivedi il piano e convalida le risorse e i component i dell'infrastruttura che verranno creati.
	4. Immettere il seguente comando per distribuire l'infrastruttura:
	`terraform apply`
	5. Quando richiesto, inserisci yes per confermare la distribuzione.
	6. Attendi il completamento della distribuzione.	

Verifica della distribuzione

Operazione	Description	Competenze richieste
Verifica la creazione di risorse.	1. Immetti il seguente comando per impostare il contesto Amazon EKS utilizzando AWS CLI: `aws eks update-ku beconfig –name <eks_cluster_name>`	DevOps ingegnere

Operazione	Description	Competenze richieste

	2. Inserisci il seguente comando per verificare il numero di pod che utilizzano CockroachDB:
	
	`kubectl get pods -n <namespace>`
	
	Di seguito è riportato un output di esempio.
	
	`NAME READY STATUS RESTARTS AGE`
	`cockroach-operator-655fbf7847-zn9v8 1/1 Running 0 30m`
	`cockroachdb-0 1/1 Running 0 24m`
	`cockroachdb-1 1/1 Running 0 24m`
	`cockroachdb-2 1/1 Running 0 24m`
	
	3. Verifica che il numero di pod corrisponda al valore definito nel file variable.tf.	

Operazione	Description	Competenze richieste
(Facoltativo) Scalabilità verso l'alto o verso il basso.	1. Nel file variable.tf, aumentate o diminuite il numero di nodi, quindi salvate il file. 2. Ripeti i passaggi per implementare l'infrast ruttura tramite Terraform . Terraform aggiunge o rimuove i pod. 3. Ripeti i passaggi per verificare il numero di pod che utilizzano Cockroach DB. Ad esempio, se hai aumentato il numero di nodi da tre a quattro, ora dovresti vedere quattro pod in esecuzione.	DevOps ingegnere, Terraform

Eliminazione

Operazione	Description	Competenze richieste
Eliminare l'infrastruttura.	La scalabilità dei nodi `0` ridurrà i costi di elaborazione. Tuttavia, continueranno a essere addebitati costi per i volumi Amazon EBS persisten ti creati da questo modulo. Per eliminare i costi di archiviaz ione, segui questi passaggi per eliminare tutti i volumi:	Terraform

Operazione	Description	Competenze richieste
	1. Inserisci il seguente comando per eliminare l'infrastruttura: `terraform destroy` 2. Quando richiesto, inserisci yes per confermare.	

risoluzione dei problemi

Problema	Soluzione
Errore durante la convalida delle credenziali del provider	Quando esegui il `destroy` comando `apply` o il comando Terraform, potresti riscontrare il seguente errore: `Error: configuring Terraform AWS Provider: error validating provider credentials: error calling sts:GetCallerIdentity: operation error STS: GetCallerIdentity, https response error StatusCode: 403, RequestID: 123456a9-fbc1-40ed-b8d8-513d0133ba7f, api error InvalidClientTokenId: The security token included in the request is invalid.` Questo errore è causato dalla scadenza del token di sicurezza per le credenziali utilizzate nella configurazione del computer locale. Per istruzioni su come risolvere l'errore, consulta Impostare e visualizzare le impostazioni di configurazione nella AWS CLI documentazione.

Problema	Soluzione
I pod di CockroachDB sono in sospeso	1. Grazie alla regola di pianificazione `podAntiAffinity` Kubernetes, è possibile pianificare un solo pod Cockroach DB in un nodo Amazon EKS. Se il numero di pod CockroachDB supera il numero di nodi Amazon EKS disponibili, i pod Cockroach DB potrebbero essere in sospeso. In tal caso, devi implementare Cluster Autoscaler o Karpenter in modo che il nodo Amazon EKS si ridimensioni automaticamente. Per ulteriori informazioni, consulta Scale cluster compute with Karpenter e Cluster Autoscaler. 2. Verifica se i nodi di lavoro Kubernetes hanno etichette inserendo il seguente comando: `node=cockroachdb` `kubectl get nodes --show-labels` Se non lo sono, devi assicurarti che tutti i nodi di lavoro siano etichettati correttamente. Per ulteriori informazioni, consulta Modificare la configurazione di un gruppo di nodi nella documentazione di Amazon EKS.

Risorse correlate

- Distribuisci CockroachDB in un singolo cluster Kubernetes (documentazione CockroachDB)
- Orchestra CockroachDB su più cluster Kubernetes (documentazione CockroachDB)
- AWS Provider (documentazione Terraform)

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Implementa un microservizio Java di esempio su Amazon EKS ed esponi il microservizio utilizzando un Application Load Balancer

Vijay Thompson e Akkamahadevi Hiremath, Amazon Web Services

Riepilogo

Questo modello descrive come distribuire un microservizio Java di esempio come applicazione containerizzata su Amazon Elastic Kubernetes Service (Amazon EKS) utilizzando `eksctl` l'utilità da riga di comando e Amazon Elastic Container Registry (Amazon ECR). È possibile utilizzare un Application Load Balancer per bilanciare il carico del traffico dell'applicazione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- La versione 1.7 o successiva dell'interfaccia a riga di comando AWS (AWS CLI), installata e configurata su macOS, Linux o Windows
- Un demone Docker in esecuzione
- L'utilità da riga di `eksctl` comando, installata e configurata su macOS, Linux o Windows (per ulteriori informazioni, consulta la sezione Guida introduttiva ad Amazon EKS — eksctl nella documentazione di Amazon EKS).
- L'utilità da riga di `kubectl` comando, installata e configurata su macOS, Linux o Windows (per ulteriori informazioni, consulta Installazione o aggiornamento di kubectl nella documentazione di Amazon EKS).

Limitazioni

- Questo modello non copre l'installazione di un certificato SSL per Application Load Balancer.

Architecture

Stack tecnologico Target

- Amazon ECR

- Amazon EKS
- Elastic Load Balancing

Architettura di destinazione

Il diagramma seguente mostra un'architettura per la containerizzazione di un microservizio Java su Amazon EKS.

Tools (Strumenti)

- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire Kubernetes su AWS senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.
- Elastic Load Balancing distribuisce automaticamente il traffico in entrata su più destinazioni, come istanze Amazon Elastic Compute Cloud (EC2Amazon), contenitori e indirizzi IP, in una o più zone di disponibilità.
- eksctl ti aiuta a creare cluster su Amazon EKS.
- kubectl consente di eseguire comandi contro i cluster Kubernetes.
- Docker ti aiuta a creare, testare e distribuire applicazioni in pacchetti chiamati contenitori.

Epiche

Crea un cluster Amazon EKS utilizzando eksctl

Operazione	Description	Competenze richieste
Crea un cluster Amazon EKS.	Per creare un cluster Amazon EKS che utilizza due EC2 istanze Amazon t2.small come nodi, esegui il seguente comando:	Sviluppatore, amministratore di sistema

Operazione	Description	Competenze richieste
	`eksctl create cluster --name <your-cluster-name> --version <version-number> --nodes=1 --node-type=t2.small` **Note** Il processo può richiedere dai 15 ai 20 minuti. Dopo la creazione del cluster, la configurazione Kubernetes appropriata viene aggiunta al file kubeconfig. Puoi utilizzare il `kubeconfig` file con per distribuire l'applicazione nei `kubectl` passaggi successivi.	
Verifica il cluster Amazon EKS.	Per verificare che il cluster sia stato creato e che tu possa connetterti ad esso, esegui il `kubectl get nodes` comando.	Sviluppatore, amministratore di sistema

Crea un repository Amazon ECR e invia l'immagine Docker.

Operazione	Description	Competenze richieste
Crea un repository Amazon ECR.	Segui le istruzioni riportate nella sezione Creazione di	Sviluppatore, amministratore di sistema

Operazione	Description	Competenze richieste
	un repository privato nella documentazione di Amazon ECR.	
Crea un file XML POM.	Crea un `pom.xml` file basato sul codice del file POM di esempio nella sezione Informazioni aggiuntive di questo modello.	Sviluppatore, amministratore di sistema

Operazione	Description	Competenze richieste

Crea un file sorgente.	Crea un file sorgente chiamato `HelloWorld.java` nel `src/main/java/eksExample` percorso in base al seguente esempio:
	`package eksExample;`
	`import static spark.Spark.get;`
	`public class HelloWorld {`
	`  public static void main(String[] args) {`
	`  get("/", (req, res) -> {`
	`  return "Hello World!";`
	`  });`
	`  }`
	`}`
	Assicuratevi di utilizzare la seguente struttura di cartelle:
	`### Dockerfile`
	`### deployment.yaml`
	`### ingress.yaml`
	`### pom.xml`
	`### service.yaml`
	`### src`
	`  ### main`
	`  ### java`
	`  ### eksExample`
	`  ### HelloWorld.java`	

Operazione	Description	Competenze richieste
Crea un Dockerfile.	Crea un file `Dockerfile` basato sul codice Dockerfil e di esempio nella sezione Informazioni aggiuntive di questo modello.	Sviluppatore, amministratore di sistema

Operazione	Description	Competenze richieste

Crea e invia l'immagine Docker.	Nella directory in cui desideri `Dockerfile` creare, taggare e inviare l'immagine ad Amazon ECR, esegui i seguenti comandi:
	`aws ecr get-login -password --region <region>| docker login --username <username> --password-stdin <account_number>.dkr.ecr.<region>.amazonaws.com`
	`docker buildx build --platform linux/amd64 -t hello-world-java:v1 .`
	`docker tag hello-world-java:v1 <account_number>.dkr.ecr.<region>.amazonaws.com/<repository_name>:v1`
	`docker push <account_number>.dkr.ecr.<region>.amazonaws.com/<repository_name>:v1`
	Note
	Modifica la regione AWS, il numero di account e i dettagli del repository nei comandi precedenti. Assicurati di annotare l'URL	

Operazione	Description	Competenze richieste
	dell'immagine per un uso successivo. **Important** Un sistema macOS con un chip M1 ha problemi a creare un'immagine compatibile con Amazon EKS in esecuzione su una piattaforma. AMD64 Per risolvere questo problema, usa docker buildx per creare un'immagine Docker che funzioni su Amazon EKS.	

Implementa i microservizi Java

Operazione	Description	Competenze richieste
Crea un file di distribuzione.	Crea un file YAML chiamato in `deployment.yaml` base al codice del file di distribuzione di esempio nella sezione Informazioni aggiuntive di questo modello.	Sviluppatore, amministratore di sistema

Operazione	Description	Competenze richieste
	Note Usa l'URL dell'immagine che hai copiato in precedenza come percorso del file di immagine per il repository Amazon ECR.	
Implementa i microservizi Java sul cluster Amazon EKS.	Per creare una distribuzione nel tuo cluster Amazon EKS, esegui il `kubectl apply -f deployment.yaml` comando.	Sviluppatore, amministratore di sistema
Verifica lo stato dei pod.	1. Per verificare lo stato dei pod, esegui il `kubectl get pods` comando. 2. Attendi che lo stato passi a Pronto.	Sviluppatore, amministratore di sistema
Crea un servizio.	1. Crea un file chiamato `service.yaml` in base al codice del file di servizio di esempio nella sezione Informazioni aggiuntive di questo modello. 2. Esegui il comando `kubectl apply -f service.yaml .`	Sviluppatore, amministratore di sistema

Operazione	Description	Competenze richieste
Installa il componente aggiuntivo AWS Load Balancer Controller.	Segui le istruzioni contenute nell'installazione del componente aggiuntivo AWS Load Balancer Controller nella documentazione di Amazon EKS. **Note** È necessario che il componente aggiuntivo sia installato per creare un Application Load Balancer o un Network Load Balancer per un servizio Kubernetes.	Sviluppatore, amministratore di sistema
Crea una risorsa di ingresso.	Crea un file YAML chiamato in `ingress.yaml` base al codice del file di risorse di ingresso di esempio nella sezione Informazioni aggiuntive di questo modello.	Sviluppatore, amministratore di sistema
Crea un Application Load Balancer.	Per distribuire la risorsa in ingresso e creare un Application Load Balancer, esegui il comando. `kubectl apply -f ingress.yaml`	Sviluppatore, amministratore di sistema

Eseguire il test dell'applicazione

Operazione	Description	Competenze richieste
Testa e verifica l'applicazione.	1. Per ottenere il nome DNS del load balancer dal campo ADDRESS, esegui il `kubectl get ingress.networking.k8s.io/java-microservice-ingress` comando. 2. Su un' EC2 istanza nello stesso VPC dei nodi Amazon EKS, esegui il `curl -v <DNS address from previous command>` comando.	Sviluppatore, amministratore di sistema

Risorse correlate

- Creazione di un repository privato (documentazione Amazon ECR)
- Inviare un'immagine Docker (documentazione Amazon ECR)
- Controller di ingresso (Amazon EKS Workshop)
- Compilazioni Docker (Docker docs)

Informazioni aggiuntive

Esempio di file POM

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/
maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
```

```
  <groupId>helloWorld</groupId>
  <artifactId>helloWorld</artifactId>
  <version>1.0-SNAPSHOT</version>


  <dependencies>
    <dependency>
      <groupId>com.sparkjava</groupId><artifactId>spark-core</
artifactId><version>2.0.0</version>
    </dependency>
  </dependencies>
  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId><artifactId>maven-jar-plugin</
artifactId><version>2.4</version>
        <configuration><finalName>eksExample</finalName><archive><manifest>
              <addClasspath>true</addClasspath><mainClass>eksExample.HelloWorld</
mainClass><classpathPrefix>dependency-jars/</classpathPrefix>
            </manifest></archive>
        </configuration>
      </plugin>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId><artifactId>maven-compiler-plugin</
artifactId><version>3.1</version>
        <configuration><source>1.8</source><target>1.8</target></configuration>
      </plugin>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId><artifactId>maven-assembly-plugin</
artifactId>
        <executions>
          <execution>
            <goals><goal>attached</goal></goals><phase>package</phase>
            <configuration>
              <finalName>eksExample</finalName>
              <descriptorRefs><descriptorRef>jar-with-dependencies</descriptorRef></
descriptorRefs>
              <archive><manifest><mainClass>eksExample.HelloWorld</mainClass></
manifest></archive>
            </configuration>
          </execution>
        </executions>
```

```
      </plugin>
    </plugins>
  </build>
</project>
```

Esempio di Dockerfile

```
FROM bellsoft/liberica-openjdk-alpine-musl:17

RUN apk add maven
WORKDIR /code

# Prepare by downloading dependencies
ADD pom.xml /code/pom.xml
RUN ["mvn", "dependency:resolve"]
RUN ["mvn", "verify"]

# Adding source, compile and package into a fat jar
ADD src /code/src
RUN ["mvn", "package"]

EXPOSE 4567
CMD ["java", "-jar", "target/eksExample-jar-with-dependencies.jar"]
```

Esempio di file di distribuzione

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: microservice-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app.kubernetes.io/name: java-microservice
  template:
    metadata:
      labels:
        app.kubernetes.io/name: java-microservice
    spec:
      containers:
      - name: java-microservice-container
        image: .dkr.ecr.amazonaws.com/:
```

```
        ports:
        - containerPort: 4567
```

File di servizio di esempio

```
apiVersion: v1
kind: Service
metadata:
  name: "service-java-microservice"
spec:
  ports:
    - port: 80
      targetPort: 4567
      protocol: TCP
  type: NodePort
  selector:
    app.kubernetes.io/name: java-microservice
```

Esempio di file di risorse di ingresso

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: "java-microservice-ingress"
  annotations:
    kubernetes.io/ingress.class: alb
    alb.ingress.kubernetes.io/load-balancer-name: apg2
    alb.ingress.kubernetes.io/target-type: ip
  labels:
    app: java-microservice
spec:
  rules:
    - http:
        paths:
          - path: /
            pathType: Prefix
            backend:
              service:
                name: "service-java-microservice"
                port:
                  number: 80
```

Implementa un'applicazione basata su gRPC su un cluster Amazon EKS e accedi ad essa con un Application Load Balancer

Kirankumar Chandrashekar e Huy Nguyen, Amazon Web Services

Riepilogo

Questo modello descrive come ospitare un'applicazione basata su gRPC su un cluster Amazon Elastic Kubernetes Service (Amazon EKS) e accedervi in modo sicuro tramite un Application Load Balancer.

gRPC è un framework RPC (Remote Procedure Call) open source che può essere eseguito in qualsiasi ambiente. Puoi usarlo per integrazioni di microservizi e comunicazioni client-server. Per ulteriori informazioni su gRPC, consulta il post sul blog di AWS Application Load Balancer support per end-to-end HTTP/2 e gRPC.

Questo modello mostra come ospitare un'applicazione basata su gRPC che viene eseguita su pod Kubernetes su Amazon EKS. Il client gRPC si connette a un Application Load Balancer tramite il protocollo HTTP/2 con una connessione crittografata. SSL/TLS L'Application Load Balancer inoltra il traffico all'applicazione gRPC in esecuzione sui pod Amazon EKS. Il numero di pod gRPC può essere ridimensionato automaticamente in base al traffico utilizzando Kubernetes Horizontal Pod Autoscaler. Il gruppo target di Application Load Balancer esegue controlli di integrità sui nodi Amazon EKS, valuta se il target è integro e inoltra il traffico solo ai nodi integri.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Docker, installato e configurato su Linux, macOS o Windows.
- AWS Command Line Interface (AWS CLI) versione 2, installata e configurata su Linux, macOS o Windows.
- eksctl, installato e configurato su Linux, macOS o Windows.
- `kubectl`, installato e configurato per accedere alle risorse sul tuo cluster Amazon EKS. Per ulteriori informazioni, consulta Installazione o aggiornamento di kubectl nella documentazione di Amazon EKS.
- g RPCurl, installato e configurato.

- Un cluster Amazon EKS nuovo o esistente. Per ulteriori informazioni, consulta Guida introduttiva ad Amazon EKS.
- Il terminale del computer è configurato per accedere al cluster Amazon EKS. Per ulteriori informazioni, consulta Configurare il computer per comunicare con il cluster nella documentazione di Amazon EKS.
- AWS Load Balancer Controller, fornito nel cluster Amazon EKS.
- Un nome host DNS esistente con un certificato o SSL valido. SSL/TLS Puoi ottenere un certificato per il tuo dominio utilizzando AWS Certificate Manager (ACM) o caricando un certificato esistente su ACM. Per ulteriori informazioni su queste due opzioni, consulta Richiesta di un certificato pubblico e Importazione di certificati in AWS Certificate Manager nella documentazione ACM.

Architecture

Il diagramma seguente mostra l'architettura implementata da questo modello.

Il diagramma seguente mostra un flusso di lavoro in cui SSL/TLS il traffico viene ricevuto da un client gRPC che viene trasferito a un Application Load Balancer. Il traffico viene inoltrato in testo semplice al server gRPC perché proviene da un cloud privato virtuale (VPC).

Tools (Strumenti)

Servizi AWS

- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella shell della riga di comando.
- Elastic Load Balancing distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni. Ad esempio, puoi distribuire il traffico tra istanze Amazon Elastic Compute Cloud (Amazon EC2), contenitori e indirizzi IP in una o più zone di disponibilità.
- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire Kubernetes su AWS senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.

Strumenti

- eksctl è un semplice strumento CLI per creare cluster su Amazon EKS.
- kubectl è un'utilità da riga di comando per eseguire comandi su cluster Kubernetes.
- AWS Load Balancer Controller ti aiuta a gestire AWS Elastic Load Balancers per un cluster Kubernetes.
- g RPCurl è uno strumento da riga di comando che consente di interagire con i servizi gRPC.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub grpc-traffic-on-alb-to-eks.

Epiche

Crea e invia l'immagine Docker del server gRPC su Amazon ECR

Operazione	Description	Competenze richieste
Crea un repository Amazon ECR.	Accedi alla Console di gestione AWS, apri la console Amazon ECR e crea un repository Amazon ECR. Per ulteriori informazioni, consulta Creazione di un repository nella documentazione di Amazon ECR. Assicurati di registrare l'URL del repository Amazon ECR. Puoi anche creare un repository Amazon ECR con AWS CLI eseguendo il seguente comando: `aws ecr create-repository --repository-name helloworld-grpc`	Amministratore cloud
Creazione dell'immagine Docker.	1. Clona il repository GitHub grpc-traffic-on-alb-to-eks.	DevOps ingegnere

Operazione	Description	Competenze richieste

	`git clone https://github.com/aws-samples/grpc-traffic-on-alb-to-eks.git`
	2. Dalla directory principale del repository, assicurati che il Dockerfile esista, quindi esegui il seguente comando per creare l'immagine Docker:
	`docker build -t <amazon_ecr_repository_url>:<Tag> .`
	Important
	Assicurati di sostituirlo `<amazon_ecr_repository_url>` con l'URL del repository Amazon ECR creato in precedenza.	

Operazione	Description	Competenze richieste

Invia l'immagine Docker ad Amazon ECR.	1. Esegui il seguente comando per accedere al repository Amazon ECR:
	`aws ecr get-login -password --region us-east-1 --no-cli-auto-prompt | docker login --username AWS --password-stdin <your_aws_account_id>.dkr.ecr.us-east-1.amazonaws.com`
	2. Invia l'immagine Docker al repository Amazon ECR eseguendo il seguente comando:
	`docker push <your_aws_account_id>.dkr.ecr.us-east-1.amazonaws.com/helloworld-grpc:1.0`
	Important
	Assicurati di sostituirlo `<your_aws_account_id>` con l'ID del tuo account AWS.
		DevOps ingegnere

Implementa i manifest Kubernetes nel cluster Amazon EKS

Operazione	Description	Competenze richieste
Modifica i valori nel file manifest di Kubernetes.	1. Modifica il file manifest `grpc-sample.yaml` Kubernetes nella cartella Kubernetes del repository in base alle tue esigenze. È necessario modificare le annotazioni e il nome host nella risorsa di ingresso. Per un esempio di risorsa in ingresso, consultate la sezione Informazioni aggiuntive. Per ulteriori informazioni sulle annotazio ni in ingresso, consulta le annotazioni in ingresso nella documentazione di Kubernetes. 2. Nella risorsa di distribuzione Kubernetes, modifica le risorse di distribuzione con l'URI (Uniform Resource Identifier) per il repositor y Amazon ECR in cui hai inviato l'immagine Docker. `image` Per un esempio di risorsa di distribuz ione, consulta la sezione Informazioni aggiuntive.	DevOps ingegnere
Distribuisci il file manifest di Kubernetes.	Distribuisci il `grpc-samp le.yaml` file nel cluster Amazon EKS eseguendo il seguente `kubectl` comando:	DevOps ingegnere

Operazione	Description	Competenze richieste
	`kubectl apply -f ./kubernetes/grpc-sample.yaml`	

Crea il record DNS per l'FQDN di Application Load Balancer

Operazione	Description	Competenze richieste
Registra il nome di dominio completo per l'Application Load Balancer.	1. Esegui il `kubectl` comando seguente per descrivere la risorsa di ingresso Kubernetes che gestisce l'Application Load Balancer: `kubectl get ingress -n grpcserver` L'output di esempio è fornito nella sezione Informazioni aggiuntive. Nell'output, il `HOSTS` campo mostra il nome host DNS per cui sono stati creati i certificati SSL. 2. Registra il nome di dominio completo (FQDN) dell'Application Load Balancer dal campo dell'`Address`output. 3. Crea un record DNS che punti all'FQDN di Applicati on Load Balancer. Se il tuo provider DNS è Amazon	DevOps ingegnere

Operazione	Description	Competenze richieste
	Route 53, puoi creare un record di alias che punti al nome di dominio completo di Application Load Balancer. Per ulteriori informazioni su questa opzione, consulta Scelta tra record alias e non alias nella documentazione di Route 53.	

Test della soluzione

Operazione	Description	Competenze richieste
Prova il server gRPC.	Usa g RPCurl per testare l'endpoint eseguendo il seguente comando: `grpcurl grpc.example.com:443 list grpc.reflection.v1alpha.ServerReflection helloworld.helloworld` **Note** `grpc.example.com` Sostituiscilo con il tuo nome DNS.	DevOps ingegnere
Prova il server gRPC utilizzando un client gRPC.	Nel client gRPC di `helloworld_client_ssl.py` esempio, sostituis	DevOps ingegnere

Operazione	Description	Competenze richieste

	ci il nome host di `grpc.example.com` con il nome host utilizzato per il server gRPC.
	
	Il seguente esempio di codice mostra la risposta del server gRPC alla richiesta del client:
	
	`python ./app/helloworld_client_ssl.py`
	`message: "Hello to gRPC server from Client"`
	
	`message: "Thanks for talking to gRPC server!! Welcome to hello world. Received message is \"Hello to gRPC server from Client\""`
	`received: true`
	
	Ciò dimostra che il client può parlare con il server e che la connessione è riuscita.	

Eliminazione

Operazione	Description	Competenze richieste
Rimuovi il record DNS.	Rimuovi il record DNS che rimanda al nome di dominio completo di Application Load Balancer creato in precedenza.	Amministratore cloud

Operazione	Description	Competenze richieste
Rimuovi il sistema di bilanciamento del carico.	Sulla console Amazon EC2, scegli Load Balancers, quindi rimuovi il load balancer creato dal controller Kubernetes per la tua risorsa in ingresso.	Amministratore cloud
Elimina il cluster Amazon EKS.	Elimina il cluster Amazon EKS utilizzando`eksctl`: `eksctl delete cluster -f ./eks.yaml`	AWS DevOps

Risorse correlate

- Bilanciamento del carico di rete su Amazon EKS
- Gruppi target per i tuoi Application Load Balancer

Informazioni aggiuntive

Esempio di risorsa di ingresso:

```
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  annotations:
    alb.ingress.kubernetes.io/healthcheck-protocol: HTTP
    alb.ingress.kubernetes.io/ssl-redirect: "443"
    alb.ingress.kubernetes.io/backend-protocol-version: "GRPC"
    alb.ingress.kubernetes.io/listen-ports: '[{"HTTP": 80}, {"HTTPS":443}]'
    alb.ingress.kubernetes.io/scheme: internet-facing
    alb.ingress.kubernetes.io/target-type: ip
    alb.ingress.kubernetes.io/certificate-arn: arn:aws:acm:<AWS-
Region>:<AccountId>:certificate/<certificate_ID>
  labels:
    app: grpcserver
    environment: dev
```

```
  name: grpcserver
  namespace: grpcserver
spec:
  ingressClassName: alb
  rules:
  - host: grpc.example.com # <----- replace this as per your host name for which the
 SSL certtficate is available in ACM
    http:
      paths:
      - backend:
          service:
            name: grpcserver
            port:
              number: 9000
        path: /
        pathType: Prefix
```

Esempio di risorsa di implementazione:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: grpcserver
  namespace: grpcserver
spec:
  selector:
    matchLabels:
      app: grpcserver
  replicas: 1
  template:
    metadata:
      labels:
        app: grpcserver
    spec:
      containers:
      - name: grpc-demo
        image: <your_aws_account_id>.dkr.ecr.us-east-1.amazonaws.com/helloworld-
grpc:1.0   #<------- Change to the URI that the Docker image is pushed to
        imagePullPolicy: Always
        ports:
        - name: grpc-api
          containerPort: 9000
        env:
```

```
        - name: POD_IP
          valueFrom:
            fieldRef:
              fieldPath: status.podIP
      restartPolicy: Always
```

Output di esempio:

```
NAME             CLASS           HOSTS                          Address
 PORTS          AGE
 grpcserver       <none>      <DNS-HostName>                 <ELB-address>
 80             27d
```

Distribuisci applicazioni containerizzate in AWS IoT Greengrass V2 esecuzione come contenitore Docker

Salih Bakir, Giuseppe Di Bella e Gustav Svalander, Amazon Web Services

Riepilogo

AWS IoT Greengrass Version 2, se distribuito come contenitore Docker, non supporta nativamente l'esecuzione di contenitori di applicazioni Docker. Questo modello mostra come creare un'immagine del contenitore personalizzata basata sull'ultima versione AWS IoT Greengrass V2 che abilita la funzionalità Docker-in-Docker (DiND). Con DiND, puoi eseguire applicazioni containerizzate all'interno dell'ambiente. AWS IoT Greengrass V2

Puoi implementare questo modello come soluzione autonoma o integrarlo con piattaforme di orchestrazione dei container come Amazon ECS Anywhere. In entrambi i modelli di implementazione, mantieni la piena AWS IoT Greengrass V2 funzionalità, comprese le funzionalità di elaborazione AWS IoT SiteWise Edge, abilitando al contempo implementazioni scalabili basate su contenitori.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Per i AWS IoT Greengrass Version 2 prerequisiti generali, vedere Prerequisiti nella AWS IoT Greengrass Version 2 documentazione.
- Docker Engine, installato e configurato su Linux, macOS o Windows.
- Docker Compose (se utilizzi l'interfaccia a riga di comando (CLI) di Docker Compose per eseguire immagini Docker).
- Un sistema operativo Linux.
- Un hypervisor con un server host che supporta la virtualizzazione.
- Requisiti di sistema:
  - 2 GB di RAM (minimo)
  - 5 GB di spazio disponibile su disco (minimo)
  - Per AWS IoT SiteWise Edge, una CPU quad-core x86_64 con 16 GB di RAM e 50 GB di spazio disponibile su disco. Per ulteriori informazioni sull'elaborazione AWS IoT SiteWise dei dati, consulta Requisiti del pacchetto di elaborazione dati nella documentazione. AWS IoT SiteWise

Versioni del prodotto

- AWS IoT Greengrass Version 2 versione 2.5.3 o successiva
- Docker-in-Docker versione 1.0.0 o successiva
- Docker Compose versione 1.22 o successiva
- Docker Engine versione 20.10.12 o successiva

Limitazioni

- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità regionale, vedi AWS Servizi per regione. Per endpoint specifici, consulta Service endpoints and quotas e scegli il link relativo al servizio.

Architecture

Stack tecnologico Target

- Fonti di dati: dispositivi IoT, sensori o apparecchiature industriali che generano dati per l'elaborazione
- AWS IoT Greengrass V2— Funzionamento come contenitore Docker con D-in-D funzionalità, distribuito su infrastrutture periferiche
- Applicazioni containerizzate: applicazioni personalizzate eseguite all'interno dell'ambiente come contenitori Docker annidati AWS IoT Greengrass V2
- (Facoltativo) Amazon ECS Anywhere: orchestrazione dei container che gestisce la distribuzione dei container AWS IoT Greengrass V2
- Altro Servizi AWS — AWS IoT Core e altro Servizi AWS per AWS IoT SiteWise l'elaborazione e la gestione dei dati

Architettura Target

Il diagramma seguente mostra un esempio di architettura di distribuzione di destinazione che utilizza Amazon ECS Anywhere, uno strumento di gestione dei container.

Il diagramma mostra il flusso di lavoro seguente:

1: Archiviazione delle immagini dei container: Amazon ECR archivia le immagini dei AWS IoT Greengrass container e tutti i contenitori di applicazioni personalizzati necessari per l'elaborazione edge.

2 e 3: Implementazione dei container: Amazon ECS Anywhere distribuisce AWS IoT Greengrass l'immagine del contenitore da Amazon ECR all'edge location, gestendo il ciclo di vita del contenitore e il processo di distribuzione.

4: Distribuzione dei componenti: il AWS IoT Greengrass core distribuito distribuisce automaticamente i componenti pertinenti in base alla configurazione. I componenti includono AWS IoT SiteWise Edge e altri componenti necessari per l'elaborazione dei bordi all'interno dell'ambiente containerizzato.

5: Inserimento dei dati: dopo la configurazione completa, AWS IoT Greengrass inizia a importare dati di telemetria e sensori da varie fonti di dati IoT presso l'edge location.

6: Elaborazione dei dati e integrazione nel cloud: il AWS IoT Greengrass core containerizzato elabora i dati localmente utilizzando i componenti implementati (incluso Edge per i dati industriali). AWS IoT SiteWise Quindi, invia i dati elaborati ai Cloud AWS servizi per ulteriori analisi e archiviazione.

Tools (Strumenti)

Servizi AWS

- Amazon ECS Anywhere ti aiuta a distribuire, utilizzare e gestire attività e servizi Amazon ECS sulla tua infrastruttura.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- AWS IoT Greengrassè un servizio cloud e runtime edge open source per l'Internet of Things (IoT) che ti aiuta a creare, implementare e gestire applicazioni IoT sui tuoi dispositivi.
- AWS IoT SiteWiseti aiuta a raccogliere, modellare, analizzare e visualizzare i dati delle apparecchiature industriali su larga scala.

Altri strumenti

- Docker è un insieme di prodotti Platform as a Service (PaaS) che utilizzano la virtualizzazione a livello di sistema operativo per fornire software in container.

- Docker Compose è uno strumento per definire ed eseguire applicazioni multi-contenitore.
- Docker Engine è una tecnologia di containerizzazione open source per la creazione e la containerizzazione di applicazioni.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub AWS IoT Greengrass v2 Docker-in-Docker.

Epiche

Costruisci l'immagine AWS IoT Greengrass V2 Docker-in-Docker

Operazione	Description	Competenze richieste

Clona e accedi al repository.	Per clonare il repository, usa il seguente comando:
	`git clone https://github.com/aws-samples/aws-iot-greengrass-v2-docker-in-docker.git`
	Per accedere alla `docker` directory, utilizzate il seguente comando:
	`cd aws-iot-greengrass-v2-docker-in-docker/docker`
		DevOps ingegnere, AWS DevOps

Creazione dell'immagine Docker.	Per creare l'immagine Docker con la versione predefinita (più recente), esegui il seguente comando:	AWS DevOps, DevOps ingegnere, sviluppatore di app

Operazione	Description	Competenze richieste

	`docker build -t x86_64/aws-iot-greengrass:latest .`
	
	Oppure, per creare l'immagine Docker con una versione specifica, esegui il seguente comando:
	
	`docker build --build-arg GREENGRASS_RELEASE_VERSION=2.12.0 -t x86_64/aws-iot-greengrass:2.12.0 .`
	
	Per verificare la build, esegui il comando seguente:
	
	`docker images | grep aws-iot-greengrass`	

Operazione	Description	Competenze richieste

(Facoltativo) Invia ad Amazon ECR.	1. Per creare il repository, esegui il seguente comando:
	`aws ecr create-repository --repository-name aws-iot-greengrass-dind --region us-east-1`
	2. Per autenticarti, esegui il seguente comando:
	`aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin <ACCOUNT-ID>.dkr.ecr.us-east-1.amazonaws.com`
	3. Per taggare e inviare push, esegui i seguenti comandi:
	`docker tag x86_64/aws-iot-greengrass:latest <ACCOUNT-ID>.dkr.ecr.us-east-1.amazonaws.com/aws-iot-greengrass-dind:latest`
	`docker push <ACCOUNT_ID>.dkr.ecr.us-east-1.amazonaws.com/aws-iot-g`
		Sviluppatore di app, AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	`reengrass-dind:lat est`	

Configura le AWS credenziali

Operazione	Description	Competenze richieste
Seleziona il metodo di autenticazione.	Scegli una delle seguenti opzioni: • Opzione 1 (consigliata): ruoli IAM: da utilizzare se in esecuzione su Amazon EC2, Amazon ECS o Amazon EKS con ruoli IAM appropriati. Non è necessari a alcuna configurazione aggiuntiva. • Opzione 2: variabili di ambiente: per test e sviluppo. • Opzione 3: file di credenzia li: non consigliato per la produzione. • Opzione 4: `env.cfg` file legacy.	Amministratore AWS
Configura il metodo di autenticazione.	Per il metodo di autentica zione selezionato, utilizza la seguente guida alla configura zione: • Opzione 2 (variabili di ambiente): preparazione `AWS_ACCESS_KEY_ID` e	Amministratore AWS

Operazione	Description	Competenze richieste

	facoltativamenteAWS_SESSION_TOKEN . AWS_SECRET_ACCESS_KEY
	• Opzione 3 (file delle credenziali): crea la directory e il file delle credenziali:
	• (a) Eseguite il seguente comando:
	`mkdir -p ~/greengrass-v2-credentials`
	• (b) Crea il seguente file su`~/greengrass-v2-credentials/credentials` :
	`[default]`
	`aws_access_key_id=YourAccessKey`
	`aws_secret_access_key=YourSecretKey`
	• (c) Proteggi il file come segue:
	`chmod 600 ~/greengrass-v2-credentials/credentials`
	• Opzione 4 (**`env.cfg`**) — Crea il `env.cfg` file nella	

Operazione	Description	Competenze richieste

	`docker` directory come segue:
	`GGC_ROOT_PATH=/greengrass/v2`
	`AWS_REGION=us-east-1`
	`PROVISION=true`
	`THING_NAME=MyGreengrassCore`
	`THING_GROUP_NAME=MyGreengrassCoreGroup`
	`TES_ROLE_NAME=GreengrassV2TokenExchangeRole`
	`TES_ROLE_ALIAS_NAME=GreengrassCoreTokenExchangeRoleAlias`
	`COMPONENT_DEFAULT_USER=ggc_user:ggc_group`	

Esegui con Docker Compose

Operazione	Description	Competenze richieste
Configurare `docker-compose.yml` .	Aggiorna il `docker-compose.yml` file con le	DevOps ingegnere

Operazione	Description	Competenze richieste

	variabili di ambiente come segue:
	• Variabili chiave: `PROVISION, AWS_REGION THING_NAME ,THING_GROUP_NAME ,TES_ROLE_NAME , TES_ROLE_ALIAS_NAME`
	• Per le credenziali tramite variabili di ambiente, decommenta quanto segue:
	`environment:`
	`- AWS_ACCESS_KEY_ID=YourKey`
	`- AWS_SECRET_ACCESS_KEY=YourSecret`
	• Per il file delle credenziali, decommenta quanto segue:
	`volumes:`
	`- ~/greengrass-v2-credentials:/root/.aws/:ro`
	• Per la persistenza dei log, aggiungete quanto segue:
	`volumes:`	

Operazione	Description	Competenze richieste
	`- ./logs:/greengrass/v2/logs`	

Avvia e verifica il contenitore.	Per iniziare in primo piano, esegui il seguente comando:
	`docker-compose up --build`
	Oppure, per iniziare in background, esegui il seguente comando:
	`docker-compose up --build -d`
	Per verificare lo stato, esegui il comando seguente:
	`docker-compose ps`
	Per monitorare i log, esegui il comando seguente:
	`docker-compose logs -f`
		DevOps ingegnere

Esegui con Docker CLI

Operazione	Description	Competenze richieste
Esegui il contenitore con la CLI Docker.	• Per avviare un'esecuzione di base senza provision ing, esegui il comando seguente:	DevOps ingegnere

Operazione	Description	Competenze richieste

	`docker run --init --privileged -it --name aws-iot-greengrass x86_64/aws-iot-greengrass:latest`
	
	• Per avviare un'esecuzione con variabili di ambiente, esegui il comando seguente:
	
	`docker run --init --privileged -it --name aws-iot-greengrass -e PROVISION=true -e AWS_REGION=us-east-1 -e AWS_ACCESS_KEY_ID=YourKey -e AWS_SECRET_ACCESS_KEY=YourSecret -e THING_NAME=MyGreengrassCore -e THING_GROUP_NAME=MyGreengrassCoreGroup -p 8883:8883 x86_64/aws-iot-greengrass:latest`
	
	• Per avviare un'esecuzione con un file di credenziali, esegui il comando seguente:	

Operazione	Description	Competenze richieste

	`docker run --init --privileged -it --name aws-iot-greengrass -v ~/greengrass-v2-credentials:/root/.aws:ro -e PROVISION=true -e AWS_REGION=us-east-1 -e THING_NAME=MyGreengrassCore -p 8883:8883 x86_64/aws-iot-greengrass:latest`
	• Per avviare un'esecuzione con un `env.cfg` file, esegui il comando seguente:
	`docker run --init --privileged -it --name aws-iot-greengrass -v ~/greengrass-v2-credentials:/root/.aws:ro --env-file env.cfg -p 8883:8883 x86_64/aws-iot-greengrass:latest`
	• Per avviare un'esecuzione con log persistenti, esegui il comando seguente:	

Operazione	Description	Competenze richieste
	`docker run --init --privileged -it --name aws-iot-greengrass -v \ $(pwd)/logs:/greengrass/v2/logs x86_64/aws-iot-greengrass:latest`	

Verifica il contenitore.	• Per verificare lo stato del contenitore, esegui il seguente comando:
	`docker ps | grep aws-iot-greengrass`
	• Per monitorare i log, esegui il seguente comando:
	`docker logs -f aws-iot-greengrass`
		DevOps ingegnere

Gestisci applicazioni containerizzate

Operazione	Description	Competenze richieste
Distribuisci applicazioni.	• Per distribuire il contenitore dell'applicazione, esegui il comando seguente: `docker run -d --name my-app nginx:alpine`	Sviluppatore di app

Operazione	Description	Competenze richieste

	• Per verificare la distribuzione, esegui il comando seguente:
	`docker ps`
	• Per accedere ai log, esegui il comando seguente:
	`cd /greengrass/v2/logs && cat greengrass.log`	

Accesso e test Docker-in-Docker.	1. Per aprire una shell nel contenitore, esegui il seguente comando:
	`docker exec -it aws-iot-greengrass sh`
	2. Per verificare Docker, esegui i seguenti comandi:
	`docker --version`
	`docker ps`
	3. Per eseguire il test Docker-in-Docker, esegui il comando seguente:
	`docker run --rm hello-world`
		DevOps ingegnere

(Opzionale) Integrazione con Amazon ECS Anywhere

Operazione	Description	Competenze richieste

Configura il cluster Amazon ECS.	• Per creare un cluster Amazon ECS, esegui il seguente comando:
	`aws ecs create-cluster --cluster-name greengrass-cluster --region us-east-1`
	• Per creare l'attivazione SSM, esegui il seguente comando:
	`aws ssm create-activation --default-instance-name greengrass-edge --iam-role GreengrassECSAnywhereRole --registration-limit 10 --region us-east-1`
	• Per installare SSM Agent su un dispositivo edge, esegui il seguente comando:
	`curl -o install.sh https://s3.amazonaws.com/aws-ssm-downloads/latest/linux_amd64/install.sh`
		Amministratore AWS

Operazione	Description	Competenze richieste

	`chmod +x install.sh`
	
	`./install.sh activation-code activation-id us-east-1`	

Operazione	Description	Competenze richieste

Implementa attività Amazon ECS.	• Crea `greengrass-task-definition.json` con una configurazione delle attività che include `containerDefinitions` with `privileged:true` , variabili di ambiente e supporti di volume per `/var/lib/docker` ogni attività. `/greengrass/v2.Register` Esegui il comando seguente:
	
	`aws ecs register-task-definition --cli-input-json file://greengrass-task-definition.json`
	
	• Per creare il servizio, esegui il comando seguente:
	
	`aws ecs create-service --cluster greengrass-cluster --service-name greengrass-service --task-definition greengrass-dind --desired-count 1 --launch-type EXTERNAL --region us-east-1`
		Amministratore AWS

Operazione	Description	Competenze richieste
	• Per verificare la distribuzione, esegui il comando seguente: `aws ecs describe-services --cluster greengrass-cluster --services greengrass-service --region us-east-1`	

Interrompi e pulisci

Operazione	Description	Competenze richieste

Fermare il contenitore.	• Per fermare il contenitore utilizzando Docker Compose, esegui i seguenti comandi:
	`docker-compose stop`
	`docker-compose down`
	`docker-compose down -v`
	• Per arrestare il contenitore (rimuove i volumi) utilizzando la CLI Docker, esegui i seguenti comandi:
	`docker stop aws-iot-greengrass`
		DevOps ingegnere

Operazione	Description	Competenze richieste

	`docker rm aws-iot-greengrass`
	• Per fermare il contenitore usando force remove, esegui il seguente comando:
	`docker rm -f aws-iot-greengrass`	

risoluzione dei problemi

Problema	Soluzione

Il contenitore non si avvia con errori di autorizzazione.	1. Per assicurarti che il contenitore funzioni con il `--privileged` flag richiesto per Docker-in-Docker, esegui il comando seguente:
	`docker run --init --privileged -it --name aws-iot-greengrass x86_64/aws-iot-greengrass:latest`
	2. Nel `docker-compose.yml` file, imposta quanto segue:
	`privileged: true`
	Warning
	`--privileged` concede privilegi estesi al contenitore.

Problema	Soluzione

Il provisioning fallisce a causa di errori nelle credenziali.	Per verificare che le credenziali siano configura te correttamente, attenersi alla seguente procedura:
	1. Controlla il ruolo IAM:
	`aws sts get-caller-identity`
	2. Verifica le variabili di ambiente:
	`docker exec -it aws-iot-g reengrass sh -c 'echo \$AWS_ACC ESS_KEY_ID'`
	3. Controlla il file delle credenziali:
	`docker exec -it aws-iot-g reengrass sh -c 'cat /root/.aw s/credentials'`
	Assicurati che le autorizzazioni IAM includano `iot:CreateThing` ,`iot:CreatePolicy` , `iot:AttachPolicy` `iam:CreateRole` , e. `iam:AttachRolePolicy`

Problema	Soluzione

Impossibile connettersi al demone Docker all'interno del contenitore.	1. Controlla lo stato del demone Docker:
	`docker exec -it aws-iot-greengrass sh -c 'ps aux | grep dockerd'`
	2. Verifica che il socket Docker esista:
	`docker exec -it aws-iot-greengrass sh -c 'ls -la /var/run/docker.sock'`
	3. Avvia manualmente se necessario:
	`docker exec -it aws-iot-greengrass sh -c 'dockerd > /var/log/docker.log 2>&1 &'`

Il contenitore esaurisce lo spazio su disco.	1. Controlla lo spazio su disco:
	`df -h`
	2. Pulisci le risorse Docker:
	`docker system prune -a --volumes`
	3. Rimuovi i vecchi artefatti Greengrass:
	`docker exec -it aws-iot-greengrass sh -c 'rm -rf /greengrass/v2/work/*  '`
	Garantire uno spazio minimo su disco: 5 GB per le operazioni di base e 50 GB per Edge AWS IoT SiteWise

Problema	Soluzione

Crea problemi.	1. Assicurati di essere nella `docker` directory:
	`cd docker && pwd`
	2. Verifica che `Dockerfile` esista:
	`ls -la Dockerfile`
	3. Verifica che Docker sia installato e funzionante:
	`docker --version && docker ps`

Problemi di connettività di rete.	1. Verifica la risoluzione DNS:
	`docker exec -it aws-iot-greengrass sh -c 'nslookup google.com'`
	2. Test di AWS connettività:
	`docker exec -it aws-iot-greengrass sh -c 'curl https://iot.us-east-1.amazonaws.com'`
	Verificate che il firewall consenta il traffico HTTPS (443) e MQTT (8883) in uscita.

Problema	Soluzione

I componenti Greengrass non vengono distribuiti.	1. Controlla i registri di Greengrass:
	`docker exec -it aws-iot-greengrass sh -c 'cat /greengrass/v2/logs/greengrass.log'`
	2. Verifica la configurazione del ruolo TES:
	`aws iot describe-role-alias --role-alias TES_ROLE_ALIAS_NAME`
	Controlla i log specifici dei componenti nella directory. `/greengrass/v2/logs/`

Il contenitore esce immediatamente dopo l'avvio.	1. Verifica la presenza di errori nei log del contenitore:
	`docker logs aws-iot-greengrass`
	2. Esegui con una shell interattiva per il debug:
	`docker run --init --privileged -it --entrypoint sh x86_64/aws-iot-greengrass:latest`
	Verifica che tutte le variabili di ambiente richieste siano impostate correttamente se. `PROVISION=true` Assicurati che il `--init` flag sia usato all'avvio del contenitore.

Risorse correlate

AWS resources

- Amazon Elastic Container Service
- Configura l'elaborazione dei dati edge per AWS IoT SiteWise modelli e asset

- Che cos'è AWS IoT Greengrass

Altre risorse

- Documentazione Docker

Informazioni aggiuntive

- Per l'elaborazione dei dati AWS IoT SiteWise Edge, Docker deve essere disponibile all'interno dell' AWS IoT Greengrass ambiente.
- Per eseguire un contenitore annidato, è necessario eseguire il AWS IoT Greengrass contenitore con credenziali di amministratore.

Distribuisci contenitori utilizzando Elastic Beanstalk

Thomas Scott e Jean-Baptiste Guillois, Amazon Web Services

Riepilogo

Sul cloud Amazon Web Services (AWS), AWS Elastic Beanstalk supporta Docker come piattaforma disponibile, in modo che i contenitori possano essere eseguiti con l'ambiente creato. Questo modello mostra come distribuire contenitori utilizzando il servizio Elastic Beanstalk. L'implementazione di questo modello utilizzerà l'ambiente del server Web basato sulla piattaforma Docker.

Per utilizzare Elastic Beanstalk per distribuire e scalare applicazioni e servizi Web, devi caricare il codice e la distribuzione viene gestita automaticamente. Sono inclusi anche il provisioning della capacità, il bilanciamento del carico, la scalabilità automatica e il monitoraggio dello stato delle applicazioni. Quando usi Elastic Beanstalk, puoi assumere il pieno controllo delle risorse AWS che crea per tuo conto. Non sono previsti costi aggiuntivi per l'utilizzo di Elastic Beanstalk. Paghi solo per le risorse AWS utilizzate per archiviare ed eseguire le tue applicazioni.

Questo modello include istruzioni per la distribuzione utilizzando l'interfaccia a riga di comando AWS Elastic Beanstalk (EB CLI) e la Console di gestione AWS.

Casi d'uso

I casi d'uso di Elastic Beanstalk includono quanto segue:

- Implementa un ambiente prototipo per dimostrare un'applicazione frontend. (Questo modello utilizza un Dockerfile come esempio.)
- Implementa un'API per gestire le richieste API per un determinato dominio.
- Implementa una soluzione di orchestrazione utilizzando Docker-Compose (non `docker-compose.yml` viene utilizzato come esempio pratico in questo modello).

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS
- AWS EB CLI installata localmente
- Docker installato su un computer locale

Limitazioni

- Il piano gratuito prevede un limite Docker di 100 pull per 6 ore per indirizzo IP.

Architecture

Stack tecnologico Target

- Istanze Amazon Elastic Compute Cloud (Amazon EC2)
- Gruppo di sicurezza
- Application Load Balancer
- Gruppo Auto Scaling

Architettura Target

Automazione e scalabilità

AWS Elastic Beanstalk può scalare automaticamente in base al numero di richieste effettuate. Le risorse AWS create per un ambiente includono un Application Load Balancer, un gruppo Auto Scaling e una o più istanze Amazon. EC2

Il sistema di bilanciamento del carico si trova davanti EC2 alle istanze Amazon, che fanno parte del gruppo Auto Scaling. Amazon EC2 Auto Scaling avvia automaticamente EC2 istanze Amazon aggiuntive per far fronte all'aumento del carico sull'applicazione. Se il carico sull'applicazione diminuisce, Amazon EC2 Auto Scaling interrompe le istanze, ma mantiene almeno un'istanza in esecuzione.

Trigger di scalabilità automatica

Il gruppo Auto Scaling nel tuo ambiente Elastic Beanstalk utilizza due CloudWatch allarmi Amazon per avviare le operazioni di scalabilità. I trigger predefiniti eseguono il dimensionamento quando la media del traffico di rete in uscita da ciascuna istanza è superiore a 6 MB o inferiore a 2 MB per un intervallo di tempo di cinque minuti. Per utilizzare Amazon EC2 Auto Scaling in modo efficace, configura i trigger appropriati per la tua applicazione, il tipo di istanza e i requisiti di servizio. È possibile effettuare il dimensionamento in base a diverse statistiche, tra cui latenza, I/O su disco, uso della CPU e numero di richieste. Per ulteriori informazioni, consulta Trigger di Auto Scaling.

Tools (Strumenti)

Servizi AWS

- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.
- AWS EB Command Line Interface (EB CLI) è un client a riga di comando che puoi utilizzare per creare, configurare e gestire ambienti Elastic Beanstalk.
- Elastic Load Balancing distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni. Ad esempio, puoi distribuire il traffico tra istanze Amazon Elastic Compute Cloud (Amazon EC2), contenitori e indirizzi IP in una o più zone di disponibilità.

Altri servizi

- Docker impacchetta il software in unità standardizzate chiamate contenitori che includono librerie, strumenti di sistema, codice e runtime.

Codice

Il codice per questo pattern è disponibile nel repository GitHub Cluster Sample Application.

Epiche

Costruisci con un Dockerfile

Operazione	Description	Competenze richieste
Clonare il repository remoto.	• Per clonare il repository, esegui il comando. `git clone https://github.com/aws-samples/cluster-sample-app.git < /p >`	Sviluppatore di app, amministratore AWS, AWS DevOps
Inizializza il progetto Elastic Beanstalk Docker.	1. Crea un file chiamato alla radice. `aws.json` 2. Nel `aws.json` file, aggiungi il codice seguente.	Sviluppatore di app, amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste

	<pre>{
	"AWSEBDoc
	kerrunVersion":"1",
	"Image":{
	"Name":"c
	luster-sample-app"
	},
	"Ports":[
	{
	"ContainerPort":80
	,
	"HostPort":8080
	}
	]
	}</pre> 3. Esegui il comando `eb init -p docker` alla radice del progetto.	

Testa il progetto localmente.	1. Esegui il comando `eb local run` alla radice del progetto. 2. Prova l'applicazione accedendo a. `http://localhost`	Sviluppatore di app, amministratore AWS, AWS DevOps

Implementazione tramite EB CLI

Operazione	Description	Competenze richieste
Esegui il comando di distribuzione	1. Esegui il comando `eb create docker-sa`	Sviluppatore di app, amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste
	`mple-cluster-app` alla radice del progetto.	
Accedi alla versione distribui ta.	Al termine del comando di distribuzione, accedi al progetto utilizzando il `eb open` comando.	Sviluppatore di app, amministr atore AWS, AWS DevOps

Esegui la distribuzione utilizzando la console

Operazione	Description	Competenze richieste
Distribuisci l'applicazione utilizzando il browser.	1. Apri la console. 2. Vai alla console Elastic Beanstalk. 3. Scegli Crea applicazione. 4. Per il nome dell'appl icazione, immettete Cluster-Sample-App. 5. Scegli Docker come piattaforma. 6. Scegli Carica il tuo codice. 7. Scegli il tuo file.zip locale (nella radice del progetto clonato) o un URL pubblico di Amazon Simple Storage Service (Amazon S3).	Sviluppatore di app, amministr atore AWS, AWS DevOps
Accedi alla versione distribui ta.	Dopo la distribuzione, accedi all'applicazione distribuita e scegli l'URL fornito.	Sviluppatore di app, amministr atore AWS, AWS DevOps

Risorse correlate

- Ambienti di server Web
- Installare l'EB CLI su macOS
- Installazione manuale dell'EB CLI

Informazioni aggiuntive

Vantaggi dell'utilizzo di Elastic Beanstalk

- Provisioning automatico dell'infrastruttura
- Gestione automatica della piattaforma sottostante
- Patch e aggiornamenti automatici per supportare l'applicazione
- Ridimensionamento automatico dell'applicazione
- Possibilità di personalizzare il numero di nodi
- Possibilità di accedere ai componenti dell'infrastruttura, se necessario
- Facilità di implementazione rispetto ad altre soluzioni di implementazione di container

Genera un indirizzo IP statico in uscita utilizzando una funzione Lambda, Amazon VPC e un'architettura serverless

Thomas Scott, Amazon Web Services

Riepilogo

Questo modello descrive come generare un indirizzo IP statico in uscita nel cloud Amazon Web Services (AWS) utilizzando un'architettura serverless. La tua organizzazione può trarre vantaggio da questo approccio se desidera inviare file a un'entità aziendale separata utilizzando il Secure File Transfer Protocol (SFTP). Ciò significa che l'entità aziendale deve avere accesso a un indirizzo IP che consenta ai file di attraversare il firewall.

L'approccio del pattern ti aiuta a creare una funzione AWS Lambda che utilizza un indirizzo IP elastico come indirizzo IP in uscita. Seguendo i passaggi di questo schema, è possibile creare una funzione Lambda e un cloud privato virtuale (VPC) che instrada il traffico in uscita attraverso un gateway Internet con un indirizzo IP statico. Per utilizzare l'indirizzo IP statico, si collega la funzione Lambda al VPC e alle relative sottoreti.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Autorizzazioni AWS Identity and Access Management (IAM) per creare e distribuire una funzione Lambda e per creare un VPC e le relative sottoreti. Per ulteriori informazioni su questo argomento, consulta Ruolo di esecuzione e autorizzazioni utente nella documentazione di AWS Lambda.
- Se prevedi di utilizzare l'infrastruttura come codice (IaC) per implementare l'approccio di questo modello, hai bisogno di un ambiente di sviluppo integrato (IDE) come AWS Cloud9. Per ulteriori informazioni su questo argomento, consulta Cos'è AWS Cloud9? nella documentazione di AWS Cloud9.

Architecture

Il diagramma seguente mostra l'architettura serverless per questo modello.

Il diagramma mostra il flusso di lavoro seguente:

1. Il traffico in uscita esce. `NAT gateway 1` `Public subnet 1`
2. Il traffico in uscita esce`NAT gateway 2`. `Public subnet 2`
3. La funzione Lambda può essere eseguita in `Private subnet 1` o. `Private subnet 2`
4. `Private subnet 1`e `Private subnet 2` indirizzano il traffico verso i gateway NAT nelle sottoreti pubbliche.
5. I gateway NAT inviano il traffico in uscita al gateway Internet dalle sottoreti pubbliche.
6. I dati in uscita vengono trasferiti dal gateway Internet al server esterno.

Stack tecnologico

- Lambda
- Amazon Virtual Private Cloud (Amazon VPC) (Amazon VPC)

Automazione e scalabilità

È possibile garantire l'alta disponibilità (HA) utilizzando due sottoreti pubbliche e due private in diverse zone di disponibilità. Anche se una zona di disponibilità diventa non disponibile, la soluzione del pattern continua a funzionare.

Tools (Strumenti)

- AWS Lambda: AWS Lambda è un servizio di elaborazione che supporta l'esecuzione di codice senza effettuare il provisioning o la gestione di server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo. Verrà addebitato soltanto il tempo di calcolo consumato e non verrà addebitato alcun costo quando il codice non è in esecuzione.
- Amazon VPC: Amazon Virtual Private Cloud (Amazon VPC) fornisce una sezione logicamente isolata del cloud AWS in cui è possibile avviare le risorse AWS in una rete virtuale definita dall'utente. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.

Epiche

Creazione di un nuovo VPC

Operazione	Description	Competenze richieste
Crea un nuovo VPC.	Accedi alla Console di gestione AWS, apri la console Amazon VPC e crea un VPC denominato `Lambda VPC` che abbia `10.0.0.0/25` come intervallo CIDR. IPv4 Per ulteriori informazioni sulla creazione di un VPC, consulta la sezione Guida introduttiva ad Amazon VPC nella documentazione di Amazon VPC.	Amministratore AWS

Crea due sottoreti pubbliche

Operazione	Description	Competenze richieste
Crea la prima sottorete pubblica.	1. Sulla console Amazon VPC, scegli Subnet, quindi scegli Crea sottorete. 2. Per Nome tag, inserisci `public-one` . 3. Per VPC, scegliere `Lambda VPC`. 4. Scegli una zona di disponibilità e registrala. 5. Per il blocco IPv4 CIDR, inserisci `10.0.0.0/28` e quindi scegli Crea sottorete.	Amministratore AWS

Operazione	Description	Competenze richieste
Crea la seconda sottorete pubblica.	1. Sulla console Amazon VPC, scegli Subnet, quindi scegli Crea sottorete. 2. Per Nome tag, inserisci `public-two` . 3. Per VPC, scegliere `Lambda VPC`. 4. **Important** Scegli una zona di disponibilità e registrala. : non è possibile utilizzare la zona di disponibilità che contiene la `public-one` sottorete. 5. Per il blocco IPv4 CIDR, inserisci `10.0.0.16 /28` e quindi scegli Crea sottorete.	Amministratore AWS

Crea due sottoreti private

Operazione	Description	Competenze richieste
Crea la prima sottorete privata.	1. Sulla console Amazon VPC, scegli Subnet, quindi scegli Crea sottorete. 2. Per Nome tag, inserisci `private-one` .	Amministratore AWS

Operazione	Description	Competenze richieste
	3. Per VPC, scegliere `Lambda VPC`. 4. Scegli la zona di disponibilità che contiene la `public-one` sottorete creata in precedenza. 5. Per il blocco IPv4 CIDR, inserisci `10.0.0.32/28` e quindi scegli Crea sottorete.	
Crea la seconda sottorete privata.	1. Sulla console Amazon VPC, scegli Subnet, quindi scegli Crea sottorete. 2. Per Nome tag, inserisci `private-two` . 3. Per VPC, scegliere `Lambda VPC`. 4. Scegli la stessa zona di disponibilità che contiene la `public-two` sottorete creata in precedenza. 5. Per il blocco IPv4 CIDR, inserisci `10.0.0.64/28` e quindi scegli Crea sottorete.	Amministratore AWS

Crea due indirizzi IP elastici per i tuoi gateway NAT

Operazione	Description	Competenze richieste
Crea il primo indirizzo IP elastico.	1. Sulla console Amazon VPC, scegli Elastic IPs, quindi	Amministratore AWS

Operazione	Description	Competenze richieste
	scegli Alloca nuovo indirizzo . 2. Scegli Alloca e registra l'ID di allocazione per il tuo indirizzo IP elastico appena creato. **Note** Questo indirizzo IP elastico viene utilizzato per il primo gateway NAT.	
Crea il secondo indirizzo IP elastico.	1. Sulla console Amazon VPC, scegli Elastic IPs, quindi scegli Alloca nuovo indirizzo . 2. Scegli Alloca e registra l'ID di allocazione per questo secondo indirizzo IP elastico. **Note** Questo indirizzo IP elastico viene utilizzato per il secondo gateway NAT.	Amministratore AWS

Creazione di un Internet Gateway

Operazione	Description	Competenze richieste
Creazione di un gateway Internet	1. Sulla console Amazon VPC, scegli Internet Gateways, quindi scegli Crea gateway internet. 2. Inserisci `Lambda internet gateway` come nome e poi scegli Crea gateway internet. Assicurati di registrare l'ID del gateway Internet.	Amministratore AWS
Collega il gateway Internet al VPC.	Selezionare l'Internet Gateway appena creato, quindi selezionare Actions, Attach to VPC (Operazioni, Collega al VPC).	Amministratore AWS

Crea due gateway NAT

Operazione	Description	Competenze richieste
Crea il primo gateway NAT.	1. Sulla console Amazon VPC, scegli NAT Gateway, quindi scegli Crea NAT Gateway. 2. Inserisci `nat-one` come nome del gateway NAT. 3. Scegli `public-one` come sottorete in cui creare il gateway NAT. 4. Per Tipo di connettività, scegli Pubblico.	Amministratore AWS

Operazione	Description	Competenze richieste
	5. Per Elastic IP allocation ID, scegli il primo indirizzo IP elastico creato in precedenza e associalo al gateway NAT. 6. Scegli Crea gateway NAT.	
Crea il secondo gateway NAT.	1. Sulla console Amazon VPC, scegli NAT Gateway, quindi scegli Crea NAT Gateway. 2. Inserisci `nat-two` come nome del gateway NAT. 3. Scegli `public-two` come sottorete in cui creare il gateway NAT. 4. Per Tipo di connettività, scegli Pubblico. 5. Per Elastic IP allocation ID, scegli il secondo indirizzo IP elastico creato in precedenza e associalo al gateway NAT. 6. Scegli Crea gateway NAT.	Amministratore AWS

Crea tabelle di routing per le tue sottoreti pubbliche e private

Operazione	Description	Competenze richieste
Crea la tabella di routing per la subnet public-one.	1. Sulla console Amazon VPC, scegli Route Tables, quindi scegli Create route table. 2. Inserisci `public-one-subnet` come nome della	Amministratore AWS

Operazione	Description	Competenze richieste
	tabella di percorso, quindi scegli Crea tabella di rotte. 3. Scegli la tabella dei `public-one-subnet` percorsi, scegli Modifica percorsi, quindi scegli Aggiungi percorso. 4. `0.0.0.0`Specificalo nella casella Destinazione, quindi scegli l'ID del gateway Internet nell'elenco Target. 5. Nella scheda Associazioni di sottoreti, scegli Modifica associazioni di sottorete, scegli la **`public-one`** sottorete con l'intervallo **`10.0.0.0/28`** CIDR, quindi scegli Salva associazioni. 6. Seleziona Salva modifiche.	

Operazione	Description	Competenze richieste
Crea la tabella di routing per la sottorete public-two.	1. Sulla console Amazon VPC, scegli Route Tables, quindi scegli Create route table. 2. Inserisci `public-two-subnet` come nome della tabella di percorso, quindi scegli Crea tabella di rotte. 3. Scegli la tabella dei `public-two-subnet` percorsi, scegli Modifica percorsi, quindi scegli Aggiungi percorso. 4. `0.0.0.0`Specificalo nella casella Destinazione, quindi scegli l'ID del gateway Internet nell'elenco Target. 5. Nella scheda Associazioni di sottoreti, scegli Modifica associazioni di sottorete , scegli la **`public-two`** sottorete con l'intervallo **`10.0.0.16/28`** CIDR, quindi scegli Salva associazioni. 6. Seleziona Salva modifiche.	Amministratore AWS

Operazione	Description	Competenze richieste
Crea la tabella di routing per la sottorete privata.	1. Sulla console Amazon VPC, scegli Route Tables, quindi scegli Create route table. 2. Inserisci `private-one-subnet` come nome della tabella di percorso, quindi scegli Crea tabella di rotte. 3. Scegli la tabella dei `private-one-subnet` percorsi, scegli Modifica percorsi, quindi scegli Aggiungi percorso. 4. `0.0.0.0`Specificalo nella casella Destinazione, quindi scegli il gateway NAT nella `public-one` sottorete nell'elenco Target. 5. Nella scheda Associazioni di sottoreti, scegli Modifica associazioni di sottoreti , scegli la **`private-one`** sottorete con l'intervallo **`10.0.0.32/28`** CIDR, quindi scegli Salva associazioni. 6. Seleziona Salva modifiche.	Amministratore AWS

Operazione	Description	Competenze richieste
Crea la tabella di routing per la sottorete private-two.	1. Sulla console Amazon VPC, scegli Route Tables, quindi scegli Create route table. 2. Inserisci `private-two-subnet` come nome della tabella di percorso, quindi scegli Crea tabella di rotte. 3. Scegli la tabella dei `private-two-subnet` percorsi, scegli Modifica percorsi, quindi scegli Aggiungi percorso. 4. `0.0.0.0`Specificalo nella casella Destinazione, quindi scegli il gateway NAT nella `public-two` sottorete nell'elenco Target. 5. Nella scheda Associazioni di sottoreti, scegli Modifica associazioni di sottoreti , scegli la **`private-two`** sottorete con l'intervallo **`10.0.0.64/28`** CIDR, quindi scegli Salva associazioni. 6. Seleziona Salva modifiche.	Amministratore AWS

Crea la funzione Lambda, aggiungila al VPC e testa la soluzione

Operazione	Description	Competenze richieste
Crea una nuova funzione Lambda.	1. Apri la console AWS Lambda e scegli Crea funzione. 2. In Informazioni di base, inserisci il **`Lambda test`** nome della funzione, quindi scegli la lingua che preferisci in Runtime. 3. Scegli Crea funzione.	Amministratore AWS
Aggiungi la funzione Lambda al tuo VPC.	1. Nella console AWS Lambda, scegli Funzioni, quindi scegli la funzione che hai creato in precedenza. 2. Scegliere Configuration (Configurazione) e quindi scegliere VPC. 3. Scegli Modifica, quindi scegli entrambe `Lambda` VPC le sottoreti private. 4. Scegli il gruppo di sicurezza predefinito a scopo di test, quindi scegli Salva.	Amministratore AWS
Scrivi codice per chiamare un servizio esterno.	1. Nel linguaggio di programmazione che preferisci, scrivi il codice per chiamare un servizio esterno che restituisce il tuo indirizzo IP.	Amministratore AWS

Operazione	Description	Competenze richieste
	2. Verifica che l'indirizzo IP restituito corrisponda a uno dei tuoi indirizzi IP elastici.	

Risorse correlate

- Configurazione di una funzione Lambda per accedere alle risorse in un VPC

Identifica automaticamente le immagini duplicate dei container durante la migrazione a un repository Amazon ECR

Rishabh Yadav e Rishi Singla, Amazon Web Services

Riepilogo

Il modello fornisce una soluzione automatizzata per identificare se le immagini archiviate in diversi archivi di contenitori sono duplicate. Questo controllo è utile quando prevedi di migrare immagini da altri repository di container ad Amazon Elastic Container Registry (Amazon ECR).

Per informazioni di base, il modello descrive anche i componenti di un'immagine del contenitore, come image digest, manifest e tag. Quando pianifichi una migrazione ad Amazon ECR, potresti decidere di sincronizzare le immagini dei container tra i registri dei container confrontando i digest delle immagini. Prima di migrare le immagini dei container, devi verificare se tali immagini esistono già nel repository Amazon ECR per evitare la duplicazione. Tuttavia, può essere difficile rilevare la duplicazione confrontando i digest di immagini e ciò potrebbe causare problemi nella fase iniziale di migrazione. Questo modello confronta i digest di due immagini simili archiviate in registri di contenitori diversi e spiega perché i digest variano, per aiutarvi a confrontare le immagini in modo accurato.

Prerequisiti e limitazioni

- Un attivo Account AWS
- Accesso al registro pubblico di Amazon ECR
- Familiarità con quanto segue: Servizi AWS
  - AWS CodeCommit
  - AWS CodePipeline
  - AWS CodeBuild
  - AWS Identity and Access Management (IAM)
  - Amazon Simple Storage Service (Amazon S3)
- CodeCommit Credenziali configurate (vedi istruzioni)

Architecture

Componenti dell'immagine del contenitore

Il diagramma seguente illustra alcuni dei componenti di un'immagine contenitore. Questi componenti sono descritti dopo il diagramma.

Termini e definizioni

I seguenti termini sono definiti nella Open Container Initiative (OCI) Image Specification.

- Registry: un servizio per l'archiviazione e la gestione delle immagini.
- Client: uno strumento che comunica con i registri e funziona con immagini locali.
- Push: il processo per caricare le immagini in un registro.
- Pull: il processo per scaricare immagini da un registro.
- Blob: la forma binaria del contenuto che viene archiviata da un registro e può essere indirizzata tramite un digest.
- Indice: costrutto che identifica più manifesti di immagini per diverse piattaforme di computer (ad esempio x86-64 o ARM a 64 bit) o tipi di media. Per ulteriori informazioni, vedere la specifica OCI Image Index.
- Manifest: un documento JSON che definisce un'immagine o un artefatto che viene caricato tramite l'endpoint del manifesto. Un manifest può fare riferimento ad altri blob in un repository utilizzando descrittori. Per ulteriori informazioni, vedere la specifica OCI Image Manifest.
- Livello del filesystem: librerie di sistema e altre dipendenze per un'immagine.
- Configurazione: un blob che contiene i metadati degli artefatti e a cui viene fatto riferimento nel manifesto. Per ulteriori informazioni, vedere la specifica di configurazione dell'immagine OCI.
- Oggetto o artefatto: un contenuto concettuale archiviato come blob e associato a un manifesto di accompagnamento con una configurazione.
- Digest: un identificatore univoco creato da un hash crittografico del contenuto di un manifesto. L'image digest aiuta a identificare in modo univoco un'immagine di contenitore immutabile. Quando estraete un'immagine utilizzando il relativo digest, scaricherete sempre la stessa immagine su qualsiasi sistema operativo o architettura. Per ulteriori informazioni, consulta la specifica dell'immagine OCI.
- Tag: un identificatore di manifesto leggibile dall'uomo. Rispetto ai digest di immagini, che sono immutabili, i tag sono dinamici. Un tag che punta a un'immagine può cambiare e spostarsi da un'immagine all'altra, sebbene l'image digest sottostante rimanga lo stesso.

Architettura di Target

Il diagramma seguente mostra l'architettura di alto livello della soluzione fornita da questo modello per identificare immagini di container duplicate confrontando le immagini archiviate in Amazon ECR e negli archivi privati.

Tools (Strumenti)

Servizi AWS

- CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita in tutte le regioni. Account AWS
- AWS CodeBuildè un servizio di compilazione completamente gestito che consente di compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per l'implementazione.
- AWS CodeCommitè un servizio di controllo delle versioni che consente di archiviare e gestire in modo privato gli archivi Git, senza dover gestire il proprio sistema di controllo del codice sorgente.
- AWS CodePipelineti aiuta a modellare e configurare rapidamente le diverse fasi di una versione del software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.

Codice

Il codice di questo modello è disponibile nel GitHub repository Soluzione automatizzata per identificare immagini di container duplicate tra i repository.

Best practice

- CloudFormation migliori pratiche
- AWS CodePipeline migliori pratiche

poemi epici

Estrai immagini dei container dagli archivi pubblici e privati di Amazon ECR

Operazione	Description	Competenze richieste

Estrai un'immagine dall'archivio pubblico di Amazon ECR.	Dal terminale, esegui il seguente comando per estrarre l'immagine `amazonlinux` dall'archivio pubblico di Amazon ECR.
	`$~ % docker pull public.ecr.aws/amazonlinux/amazonlinux:2018.03`
	Quando l'immagine è stata caricata sul tuo computer locale, vedrai il seguente pull digest, che rappresenta l'indice dell'immagine.
	`2018.03: Pulling from amazonlinux/amazonlinux`
	`4ddc0f8d367f: Pull complete`
	`Digest: sha256:f972d24199508c52de7ad37a298bda35d8a1bd7df158149b381c03f6c6e363b5`
	`Status: Downloaded newer image for public.ecr.aws/amazonlinux/amazonlinux:2018.03`
		Sviluppatore di app, AWS DevOps, amministratore AWS

Operazione	Description	Competenze richieste
	`public.ecr.aws/amazonlinux/amazonlinux:2018.03`	

Operazione	Description	Competenze richieste

Invia l'immagine a un repository privato Amazon ECR.	1. Crea un repository Amazon ECR privato denominato `test_ecr_repository` nella regione Stati Uniti orientali (Virginia settentrionale) (). `us-east-1`
	`$~ % aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin <account-id>.dkr.ecr.us-east-1.amazonaws.com`
	`Login Succeeded`
	dove si `<account-id>` riferisce al tuo. Account AWS
	2. Etichetta l'immagine locale che hai estratto in precedenza. Usa il valore `public.ecr.aws/amazonlinux/amazonlinux:2018.03` e trasferisci lo nell'archivio privato di Amazon ECR.
	`$~ % docker tag public.ecr.aws/amazonlinux/amazonlinux:2018.03 <account-id>.dkr.ecr.us-east-1.amazonaws.com/test_ecr_repository:latest`
		Amministratore AWS, AWS DevOps, sviluppatore di app

Operazione	Description	Competenze richieste

	`$~ % docker push <account-id>.dkr.ecr.us-east-1.amazonaws.com/test_ecr_repository:latest`
	
	Quando invii l'immagine al repository Amazon ECR, Docker invierà l'immagine sottostante e non l'indice dell'immagine.
	
	`The push refers to repository [<account-id>.dkr.ecr.us-east-1.amazonaws.com/test_ecr_repository] d5655967c2c4: Pushed latest: digest: sha256:52db9000073d93b9bdee6a7246a68c35a741aaade05a8f4febba0bf795cdac02 size: 529`	

Operazione	Description	Competenze richieste

Estrai la stessa immagine dall'archivio privato di Amazon ECR.	1. Dal terminale, esegui il seguente comando per estrarre l'immagine precedentemente inserita nell'archivio privato di Amazon ECR.
	
	`$~ % docker pull <account-id>.dkr.ecr.us-east-1.amazonaws.com/test_ecr_repository:latest`
	`latest: Pulling from test_ecr_repository`
	`Digest: sha256:52db9000073d93b9bdee6a7246a68c35a741aaade05a8f4febba0bf795cdac02`
	`Status: Image is up to date for <account-id>.dkr.ecr.us-east-1.amazonaws.com/test_ecr_repository:latest`
	`<account-id>.dkr.ecr.us-east-1.amazonaws.com/test_ecr_repository:latest`
	
	Il digest di questa immagine corrisponde al digest dell'immagine che hai inviato all'archivio privato di Amazon ECR e rappresenta l'immagine sottostante. Questo valore non corrisponde all'indice
		Sviluppatore di app, AWS DevOps, amministratore AWS

Operazione	Description	Competenze richieste

	dell'immagine che hai estratto dall'archivio pubblico.
	2. Per verificare, recupera l'indice delle immagini tramite digest.
	
	`curl -k -H "Authorization: Bearer $TOKEN" https://public.ecr.aws/v2/amazonlinux/amazonlinux/manifests/sha256:f972d24199508c52de7ad37a298bda35d8a1bd7df158149b381c03f6c6e363b55`
	`{ "schemaVersion": 2, "mediaType": "application/vnd.docker.distribution.manifest.list.v2+json", "manifests": [ { "mediaType": "application/vnd.docker.distribution.manifest.v2+json", "size": 529, "digest": "sha256:52db9000073d93b9bdee6a7246a68c35a741aaade05a8f4febba0bf795cdac02", "platform": {`	

Operazione	Description	Competenze richieste
	"architec ture": "amd64", "os": "linux" } }] }	

Confronta i manifesti delle immagini

Operazione	Description	Competenze richieste
Trova il manifesto dell'immagine archiviata nell'archivio pubblico di Amazon ECR.	Dal terminale, esegui il comando seguente per estrarre il manifesto dell'immagine `public.ecr.aws/amazonlinux/amazonlinux:2018.03` dall'archivio pubblico di Amazon ECR. $~ % docker manifest inspect public.ec r.aws/amazonlinux/ amazonlinux:2018.03 { "schemaVersion": 2, "mediaType": "application/vnd.d ocker.distribution .manifest.list.v2+ json", "manifests": [{ "mediaType": "application/vnd.d ocker.distribution .manifest.v2+json",	Amministratore AWS, AWS DevOps, sviluppatore di app

Operazione	Description	Competenze richieste

	"size": 529,
	"digest":
	"sha256:52db900007
	3d93b9bdee6a7246a6
	8c35a741aaade05a8f
	4febba0bf795cdac02",
	"platform": {
	"architec
	ture": "amd64",
	"os":
	"linux"
	}
	}
	]
	}	

Operazione	Description	Competenze richieste

Trova il manifesto dell'immagine archiviata nell'archivio privato di Amazon ECR.	Dal terminale, esegui il seguente comando per estrarre il manifesto dell'immagine `<account-id>.dkr.ecr.us-east-1.amazonaws.com/test_ecr_repository:latest` dall'archivio privato di Amazon ECR.
	
	`$~ % docker manifest inspect <account-id>.dkr.ecr.us-east-1.amazonaws.com/test_ecr_repository:latest`
	
	`{ "schemaVersion": 2, "mediaType": "application/vnd.docker.distribution.manifest.v2+json", "config": { "mediaType": "application/vnd.docker.container.image.v1+json", "size": 1477, "digest": "sha256:f7cee5e1af28ad4e147589c474d399b12d9b551ef4c3e11e02d982fce5eebc68" }, "layers": [ {`
		AWS DevOps, amministratore di sistema AWS, sviluppatore di app

Operazione	Description	Competenze richieste
	"mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip", "size": 62267075, "digest": "sha256:4ddc0f8d367f424871a060e2067749f32bd36a91085e7140dcb159952f2d71453" }] }	

Operazione	Description	Competenze richieste
Confronta il digest estratto da Docker con il manifest digest per l'immagine nel repository privato di Amazon ECR.	Un'altra domanda è perché il digest fornito dal comando docker pull differisce dal digest del manifest per l'immagin e. `<account-id>.dkr.ecr.us-east-1.amazonaws.com/test_ecr_repository:latest` Il digest utilizzato per docker pull rappresenta il digest del manifesto dell'immagine, che è archiviato in un registro. Questo digest è considerato la radice di una catena di hash, poiché il manifest contiene l'hash del contenuto che verrà scaricato e importato in Docker. L'ID dell'immagine utilizzato in Docker può essere trovato in questo manifesto come. `config.digest` Rappresenta la configurazione dell'immagine utilizzata da Docker. Quindi si potrebbe dire che il manifesto è la busta e l'immagine è il contenuto della busta. Il manifest digest è sempre diverso dall'ID dell'immagine. Tuttavia, un manifesto specifico dovrebbe sempre produrre lo stesso ID di immagine. Poiché il	AWS DevOps, amministratore di sistema AWS, sviluppatore di app

Operazione	Description	Competenze richieste
	manifest digest è una catena di hash, non possiamo garantire che sarà sempre la stessa per un determinato ID di immagine. Nella maggior parte dei casi, produce lo stesso digest, sebbene Docker non possa garantirl o. La possibile differenza nel manifest digest deriva dal fatto che Docker non memorizza localmente i blob compressi con gzip. Pertanto, l'esporta zione dei livelli potrebbe produrre un digest diverso, sebbene il contenuto non compresso rimanga lo stesso. L'ID dell'immagine verifica che il contenuto non compresso sia lo stesso; in altre parole, l'ID dell'immagine è ora un identificatore indirizzabile al contenuto (). `chainID` Per confermare queste informazioni, puoi confrontare l'output del comando docker inspect sui repository pubblici e privati di Amazon ECR: 1. Esegui il seguente comando dal tuo terminale per l'immagine archiviat a nell'archivio pubblico di Amazon ECR.	

Operazione	Description	Competenze richieste

	`$~ % docker inspect public.ecr.aws/amazonlinux/amazonlinux:2018.03`
	
	Per l'output del comando, consulta la sezione Informazioni aggiuntive.
	
	2. Esegui il seguente comando dal tuo terminale per l'immagine archiviata nell'archivio privato di Amazon ECR.
	
	`$~ % docker inspect <account-id>.dkr.ecr.us-east-1.amazonaws.com/test_ecr_repository:latest`
	
	Per l'output del comando, consulta la sezione Informazioni aggiuntive.
	
	I risultati verificano che entrambe le immagini abbiano lo stesso Image ID digest e lo stesso layer digest.
	
	ID: `f7cee5e1af28ad4e147589c474d399b12d9b551ef4c3e11e02d982fce5eebc68`	

Operazione	Description	Competenze richieste
	Strati: `d5655967c 2c4e8d68f8ec7cf753 218938669e6c16ac13 24303c073c736a2e2a2` Inoltre, i digest si basano sui byte dell'oggetto gestito localmente (il file locale è un tar del livello di immagine del contenitore) o sul blob inviato al server di registro. Tuttavia, quando inviate il blob a un registro, il tar viene compresso e il digest viene calcolato nel file tar compresso. Pertanto, la differenza nel valore del docker pull digest deriva dalla compressione applicata a livello di registro (Amazon ECR privato o pubblico). **Note** Questa spiegazione è specifica per l'utilizzo di un client Docker. Questo comportamento non si verificherà con altri client come nerdctl o Finch, perché non comprimono automaticamente l'immagine durante le operazioni push e pull.	

Identifica automaticamente le immagini duplicate tra gli archivi pubblici e privati di Amazon ECR

Operazione	Description	Competenze richieste
Clonare il repository.	Clona il repository Github per questo pattern in una cartella locale: `$git clone https://github.com/aws-samples/automated-solution-to-identify-duplicate-container-images-between-repositories`	Amministratore AWS, AWS DevOps
Configura una CI/CD pipeline.	Il GitHub repository include un `.yaml` file che crea uno CloudFormation stack in cui configurare una pipeline. AWS CodePipeline 1. Accedi a Console di gestione AWS e apri la console.CloudFormation 2. Crea uno stack utilizzando il `pipeline.yaml` file modello, che si trova nella `code` cartella del repository clonato. 3. Accetta o modifica i valori predefiniti dei parametri . Specificate i valori per quanto segue: • Stack name (Nome stack) • `ArtifactStoreBucketName` — Un bucket	Amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste
	S3 esistente che verrà utilizzato per archiviare gli artefatti AWS CodePipeline • `OutputBucket` — Un bucket S3 esistente che verrà utilizzato per archiviare le immagini duplicate URIs • `SourceImageFile` — Un file di testo esistente denominato `input.txt` che contiene l'immagine dell' URIs archivio pubblico che verrà confrontata con l'archivio privato Amazon ECR per rilevare la duplicazione 4. Controlla e modifica le opzioni dello stack, quindi scegli Invia per eseguire il modello. La pipeline verrà impostata in due fasi (CodeCommit e CodeBuild, come mostrato nel diagramma dell'architettura) per identificare le immagini nell'archivio privato che esistono anche nell'archivio pubblico. La pipeline è configurata con le seguenti risorse:	

Operazione	Description	Competenze richieste
	• CodePipeline per l'orchestrazione della pipeline di distribuzione. • Un CodeCommit repository per archiviare lo script bash e il file di input. Lo script bash viene utilizzato per confrontare l'immagine del contenitore IDs nei repository pubblici e privati per trovare duplicazioni. Questo controllo viene eseguito su tutti i repository specificati Account AWS in un unico. Regione AWS • Un CodeBuild progetto per richiamare lo script bash per identificare le immagini già presenti nel repository Amazon ECR. • Ruoli IAM necessari per consentire l'accesso. • Un bucket S3 per archiviare il file di output che contiene l'immagine. URIs • Un altro bucket S3 per archiviare gli artefatti. CodePipeline	

Operazione	Description	Competenze richieste
Popola il CodeCommit repository.	Per popolare il CodeCommit repository, effettuate le seguenti operazioni: 1. Apri la CodeCommit console e vai al Regione AWS punto in cui hai creato lo CloudFormation stack. 2. Trova il repository di cui hai effettuato il provisioning utilizzando lo CloudFormation script dall'elenco, scegli Clona URL, quindi copia il protocollo URL HTTPS per connetterti al repository. 3. Apri un prompt dei comandi ed esegui il comando git clone con l'URL HTTPS che hai copiato nel passaggio precedente. 4. Passa alla directory principale. Crea un file denominato `input.txt` e compila questo file con il registro delle immagini pubbliche di Amazon ECR URIs che desideri cercare nell'archivio privato di Amazon ECR. 5. Copia i file `script.sh` e `input.txt` dalla tua copia locale del repository Soluzione automatiz	Amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste

	zata per identificare le immagini duplicate dei container tra i GitHub repository all'archivio clonato. `buildspec.yml` CodeCommit
	6. Carica i file su usando questi comandi: CodeCommit
	`git add .`
	`git commit -m "added input files"`
	`git push`	

Elimina.	Per evitare di incorrere in future spese, elimina le risorse seguendo questi passaggi: 1. Vai al bucket S3 che memorizza gli CodePipeline artefatti e svuota il bucket. 2. Vai al bucket S3 che memorizza l'immagine duplicata e svuota il bucket. URIs 3. Vai alla CloudFormation console ed elimina lo stack che hai creato per configura re la pipeline.	Amministratore AWS

risoluzione dei problemi

Problema	Soluzione
Quando provi a inviare, estrarre o interagire in altro modo con un CodeCommit repository dal terminale o dalla riga di comando, ti viene richiesto di fornire un nome utente e una password e devi fornire le credenziali Git per il tuo utente IAM.	Le cause più comuni di questo errore sono le seguenti: • Nel computer locale è in esecuzione un sistema operativo che non supporta la gestione delle credenziali oppure non è installata un'utilità di gestione delle credenziali. • Le credenziali Git per il tuo utente IAM non sono state salvate in uno di questi sistemi di gestione delle credenziali. A seconda del sistema operativo in uso e dell'ambiente locale, può essere necessari o installare un sistema di gestione delle credenziali, configurare quello incluso nel sistema operativo oppure personalizzare l'ambiente locale per l'utilizzo di un sistema di storage delle credenziali. Ad esempio, se sul computer è in esecuzione macOS, è possibile utilizzare l'utilità Keychain Access per archiviare le credenziali. Se il computer esegue Windows puoi utilizzare la funzionalità Git Credential Manager installata con Git per Windows. Per ulteriori informazioni, consulta Configurazione per utenti HTTPS che utilizzano credenziali Git nella CodeCommit documentazione e Archiviazione delle credenziali nella documentazione Git.
Quando invii un'immagine al repository Amazon ECR, riscontri errori HTTP 403 o «nessuna credenziale di autenticazione di base».	Potresti riscontrare questi messaggi di errore dal comando docker push o docker pull, anche se ti sei autenticato con successo su Docker

Problema	Soluzione
	utilizzando il comando aws ecr. get-login-password Le cause note sono: • Ti sei autenticato in un'altra regione. Per ulteriori informazioni, consulta Autenticazione del registro privato nella documentazione di Amazon ECR. • Ti sei autenticato per effettuare il push su un repository per il quale non disponi delle autorizzazioni. Per ulteriori informazioni, consulta Politiche di archivio privato nella documentazione di Amazon ECR. • Il tuo token è scaduto. Il periodo di scadenza predefinito per i token ottenuti utilizzando l'`GetAuthorizationToken` operazione è di 12 ore.

Risorse correlate

- Soluzione automatizzata per identificare le immagini duplicate dei container tra i repository (repository) GitHub
- Galleria pubblica di Amazon ECR
- Immagini private in Amazon ECR (documentazione Amazon ECR)
- AWS::CodePipeline::Pipeline risorsa (documentazione)CloudFormation
- Specificazione del formato di immagine OCI

Informazioni aggiuntive

Risultato dell'ispezione Docker per l'immagine nell'archivio pubblico Amazon ECR

```
[
    {
        "Id":
 "sha256:f7cee5e1af28ad4e147589c474d399b12d9b551ef4c3e11e02d982fce5eebc68",
```

```
        "RepoTags": [
            "<account-id>.dkr.ecr.us-east-1.amazonaws.com/test_ecr_repository:latest",
            "public.ecr.aws/amazonlinux/amazonlinux:2018.03"
        ],
        "RepoDigests": [
            "<account-id>.dkr.ecr.us-east-1.amazonaws.com/
test_ecr_repository@sha256:52db9000073d93b9bdee6a7246a68c35a741aaade05a8f4febba0bf795cdac02",
            "public.ecr.aws/amazonlinux/
amazonlinux@sha256:f972d24199508c52de7ad37a298bda35d8a1bd7df158149b381c03f6c6e363b5"
        ],
        "Parent": "",
        "Comment": "",
        "Created": "2023-02-23T06:20:11.575053226Z",
        "Container":
 "ec7f2fc7d2b6a382384061247ef603e7d647d65f5cd4fa397a3ccbba9278367c",
        "ContainerConfig": {
            "Hostname": "ec7f2fc7d2b6",
            "Domainname": "",
            "User": "",
            "AttachStdin": false,
            "AttachStdout": false,
            "AttachStderr": false,
            "Tty": false,
            "OpenStdin": false,
            "StdinOnce": false,
            "Env": [
                "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
            ],
            "Cmd": [
                "/bin/sh",
                "-c",
                "#(nop) ",
                "CMD [\"/bin/bash\"]"
            ],
            "Image":
 "sha256:c1bced1b5a65681e1e0e52d0a6ad17aaf76606149492ca0bf519a466ecb21e51",
            "Volumes": null,
            "WorkingDir": "",
            "Entrypoint": null,
            "OnBuild": null,
            "Labels": {}
        },
        "DockerVersion": "20.10.17",
        "Author": "",
```

```
        "Config": {
            "Hostname": "",
            "Domainname": "",
            "User": "",
            "AttachStdin": false,
            "AttachStdout": false,
            "AttachStderr": false,
            "Tty": false,
            "OpenStdin": false,
            "StdinOnce": false,
            "Env": [
                "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
            ],
            "Cmd": [
                "/bin/bash"
            ],
            "Image":
 "sha256:c1bced1b5a65681e1e0e52d0a6ad17aaf76606149492ca0bf519a466ecb21e51",
            "Volumes": null,
            "WorkingDir": "",
            "Entrypoint": null,
            "OnBuild": null,
            "Labels": null
        },
        "Architecture": "amd64",
        "Os": "linux",
        "Size": 167436755,
        "VirtualSize": 167436755,
        "GraphDriver": {
            "Data": {
                "MergedDir": "/var/lib/docker/overlay2/
c2c2351a82b26cbdf7782507500e5adb5c2b3a2875bdbba79788a4b27cd6a913/merged",
                "UpperDir": "/var/lib/docker/overlay2/
c2c2351a82b26cbdf7782507500e5adb5c2b3a2875bdbba79788a4b27cd6a913/diff",
                "WorkDir": "/var/lib/docker/overlay2/
c2c2351a82b26cbdf7782507500e5adb5c2b3a2875bdbba79788a4b27cd6a913/work"
            },
            "Name": "overlay2"
        },
        "RootFS": {
            "Type": "layers",
            "Layers": [

 "sha256:d5655967c2c4e8d68f8ec7cf753218938669e6c16ac1324303c073c736a2e2a2"
```

```
            ]
        },
        "Metadata": {
            "LastTagTime": "2023-03-02T10:28:47.142155987Z"
        }
    }
]
```

Risultato dell'ispezione Docker per l'immagine nell'archivio privato Amazon ECR

```
[
    {
        "Id":
 "sha256:f7cee5e1af28ad4e147589c474d399b12d9b551ef4c3e11e02d982fce5eebc68",
        "RepoTags": [
            "<account-id>.dkr.ecr.us-east-1.amazonaws.com/test_ecr_repository:latest",
            "public.ecr.aws/amazonlinux/amazonlinux:2018.03"
        ],
        "RepoDigests": [
            "<account-id>.dkr.ecr.us-east-1.amazonaws.com/
test_ecr_repository@sha256:52db9000073d93b9bdee6a7246a68c35a741aaade05a8f4febba0bf795cdac02",
            "public.ecr.aws/amazonlinux/
amazonlinux@sha256:f972d24199508c52de7ad37a298bda35d8a1bd7df158149b381c03f6c6e363b5"
        ],
        "Parent": "",
        "Comment": "",
        "Created": "2023-02-23T06:20:11.575053226Z",
        "Container":
 "ec7f2fc7d2b6a382384061247ef603e7d647d65f5cd4fa397a3ccbba9278367c",
        "ContainerConfig": {
            "Hostname": "ec7f2fc7d2b6",
            "Domainname": "",
            "User": "",
            "AttachStdin": false,
            "AttachStdout": false,
            "AttachStderr": false,
            "Tty": false,
            "OpenStdin": false,
            "StdinOnce": false,
            "Env": [
                "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
            ],
            "Cmd": [
```

```
                "/bin/sh",
                "-c",
                "#(nop) ",
                "CMD [\"/bin/bash\"]"
            ],
            "Image":
"sha256:c1bced1b5a65681e1e0e52d0a6ad17aaf76606149492ca0bf519a466ecb21e51",
            "Volumes": null,
            "WorkingDir": "",
            "Entrypoint": null,
            "OnBuild": null,
            "Labels": {}
        },
        "DockerVersion": "20.10.17",
        "Author": "",
        "Config": {
            "Hostname": "",
            "Domainname": "",
            "User": "",
            "AttachStdin": false,
            "AttachStdout": false,
            "AttachStderr": false,
            "Tty": false,
            "OpenStdin": false,
            "StdinOnce": false,
            "Env": [
                "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
            ],
            "Cmd": [
                "/bin/bash"
            ],
            "Image":
"sha256:c1bced1b5a65681e1e0e52d0a6ad17aaf76606149492ca0bf519a466ecb21e51",
            "Volumes": null,
            "WorkingDir": "",
            "Entrypoint": null,
            "OnBuild": null,
            "Labels": null
        },
        "Architecture": "amd64",
        "Os": "linux",
        "Size": 167436755,
        "VirtualSize": 167436755,
        "GraphDriver": {
```

```
            "Data": {
                "MergedDir": "/var/lib/docker/overlay2/
c2c2351a82b26cbdf7782507500e5adb5c2b3a2875bdbba79788a4b27cd6a913/merged",
                "UpperDir": "/var/lib/docker/overlay2/
c2c2351a82b26cbdf7782507500e5adb5c2b3a2875bdbba79788a4b27cd6a913/diff",
                "WorkDir": "/var/lib/docker/overlay2/
c2c2351a82b26cbdf7782507500e5adb5c2b3a2875bdbba79788a4b27cd6a913/work"
            },
            "Name": "overlay2"
        },
        "RootFS": {
            "Type": "layers",
            "Layers": [

 "sha256:d5655967c2c4e8d68f8ec7cf753218938669e6c16ac1324303c073c736a2e2a2"
            ]
        },
        "Metadata": {
            "LastTagTime": "2023-03-02T10:28:47.142155987Z"
        }
    }
]
```

Installa SSM Agent sui nodi di lavoro Amazon EKS utilizzando Kubernetes DaemonSet

Mahendra Revanasiddappa, Amazon Web Services

Riepilogo

Nota, settembre 2021: l'ultima versione ottimizzata di Amazon EKS AMIs installa automaticamente SSM Agent. Per ulteriori informazioni, consulta le note di rilascio di giugno 2021 AMIs.

In Amazon Elastic Kubernetes Service (Amazon EKS), a causa delle linee guida sulla sicurezza, ai nodi di lavoro non sono associate coppie di chiavi Secure Shell (SSH). Questo modello mostra come utilizzare il tipo di DaemonSet risorsa Kubernetes per installare AWS Systems Manager Agent (SSM Agent) su tutti i nodi di lavoro, anziché installarlo manualmente o sostituire l'Amazon Machine Image (AMI) per i nodi. DaemonSet utilizza un cron job sul nodo di lavoro per pianificare l'installazione di SSM Agent. È inoltre possibile utilizzare questo modello per installare altri pacchetti sui nodi di lavoro.

Quando si risolvono i problemi del cluster, l'installazione di SSM Agent on demand consente di stabilire una sessione SSH con il nodo di lavoro, di raccogliere i log o di esaminare la configurazione dell'istanza, senza coppie di chiavi SSH.

Prerequisiti e limitazioni

Prerequisiti

- Un cluster Amazon EKS esistente con nodi di lavoro Amazon Elastic Compute Cloud (Amazon EC2).
- Le istanze di container devono disporre delle autorizzazioni necessarie per comunicare con il servizio SSM. Il ruolo gestito di AWS Identity and Access Management (IAM) Amazon SSMManaged InstanceCore fornisce le autorizzazioni necessarie per l'esecuzione di SSM Agent sulle EC2 istanze. Per ulteriori informazioni, consulta la documentazione di AWS Systems Manager.

Limitazioni

- Questo modello non è applicabile ad AWS Fargate, perché DaemonSets non sono supportati sulla piattaforma Fargate.
- Questo modello si applica solo ai nodi di lavoro basati su Linux.

- I DaemonSet pod funzionano in modalità privilegiata. Se il cluster Amazon EKS dispone di un webhook che blocca i pod in modalità privilegiata, l'agente SSM non verrà installato.

Architecture

Il diagramma seguente illustra l'architettura di questo modello.

Tools (Strumenti)

Strumenti

- kubectl è un'utilità da riga di comando utilizzata per interagire con un cluster Amazon EKS. Questo modello viene utilizzato `kubectl` per distribuire un agente SSM DaemonSet sul cluster Amazon EKS, che installerà l'agente SSM su tutti i nodi di lavoro.
- Amazon EKS semplifica l'esecuzione di Kubernetes su AWS senza dover installare, utilizzare e mantenere il tuo piano di controllo o i tuoi nodi Kubernetes. Kubernetes è un sistema open source per automatizzare l'implementazione, il dimensionamento e la gestione di applicazioni containerizzate.
- AWS Systems Manager Session Manager consente di gestire le EC2 istanze, le istanze locali e le macchine virtuali (VMs) tramite una shell interattiva basata su browser con un solo clic o tramite l'AWS Command Line Interface (AWS CLI).

Codice

Utilizza il codice seguente per creare un file di DaemonSet configurazione che installerà l'agente SSM sul cluster Amazon EKS. Segui le istruzioni nella sezione Epics.

```
cat << EOF > ssm_daemonset.yaml
apiVersion: apps/v1
kind: DaemonSet
metadata:
  labels:
    k8s-app: ssm-installer
  name: ssm-installer
  namespace: kube-system
spec:
  selector:
    matchLabels:
```

```
      k8s-app: ssm-installer
  template:
    metadata:
      labels:
        k8s-app: ssm-installer
    spec:
      containers:
      - name: sleeper
        image: busybox
        command: ['sh', '-c', 'echo I keep things running! && sleep 3600']
      initContainers:
      - image: amazonlinux
        imagePullPolicy: Always
        name: ssm
        command: ["/bin/bash"]
        args: ["-c","echo '* * * * * root yum install -y https://s3.amazonaws.com/
ec2-downloads-windows/SSMAgent/latest/linux_amd64/amazon-ssm-agent.rpm & rm -rf /etc/
cron.d/ssmstart' > /etc/cron.d/ssmstart"]
        securityContext:
          allowPrivilegeEscalation: true
        volumeMounts:
        - mountPath: /etc/cron.d
          name: cronfile
        terminationMessagePath: /dev/termination-log
        terminationMessagePolicy: File
      volumes:
      - name: cronfile
        hostPath:
          path: /etc/cron.d
          type: Directory
      dnsPolicy: ClusterFirst
      restartPolicy: Always
      schedulerName: default-scheduler
      terminationGracePeriodSeconds: 30
EOF
```

Epiche

Configura kubectl

Operazione	Description	Competenze richieste
Installa e configura kubectl per accedere al cluster EKS.	Se `kubectl` non è già installato e configurato per accedere al cluster Amazon EKS, consulta Installazione di kubectl nella documentazione di Amazon EKS.	DevOps

Implementa il DaemonSet

Operazione	Description	Competenze richieste
Crea il file DaemonSet di configurazione.	Utilizza il codice nella sezione Codice precedente di questo modello per creare un file di DaemonSet configurazione chiamato `ssm_daemonset.yaml`, che verrà distribuito nel cluster Amazon EKS. Il pod lanciato da DaemonSet ha un contenitore principale e un `init` contenitore. Il contenitore principale ha un `sleep` comando. Il `init` contenitore include una `command` sezione che crea un file cron job per installare SSM Agent sul percorso. `/etc/cron.d/` Il cron job viene eseguito solo una volta e il	DevOps

Operazione	Description	Competenze richieste
	file che crea viene automatic amente eliminato dopo il completamento del lavoro. Quando il contenitore init è terminato, il contenitore principale attende 60 minuti prima di uscire. Dopo 60 minuti, viene lanciato un nuovo pod. Questo pod installa SSM Agent, se manca, o aggiorna SSM Agent alla versione più recente. Se necessario, puoi modificar e il `sleep` comando per riavviare il pod una volta al giorno o per eseguirlo più spesso.	
Implementa il DaemonSet file sul cluster Amazon EKS.	Per distribuire il file di DaemonSet configura zione creato nel passaggio precedente sul cluster Amazon EKS, utilizza il seguente comando: `kubectl apply -f ssm_daemonset.yaml` Questo comando crea un comando DaemonSet per eseguire i pod sui nodi di lavoro per installare SSM Agent.	DevOps

Risorse correlate

- Installazione di kubectl (documentazione Amazon EKS)
- Configurazione di Session Manager (documentazione di AWS Systems Manager)

Installa l'agente SSM e l' CloudWatch agente sui nodi di lavoro Amazon EKS utilizzando preBootstrapCommands

Akkamahadevi Hiremath, Amazon Web Services

Riepilogo

Questo modello fornisce esempi di codice e passaggi per installare i nodi di lavoro AWS Systems Manager (SSM Agent) e Amazon CloudWatch sul cloud Amazon Elastic Kubernetes Service (Amazon EKS) nel cloud Amazon Web Services (AWS) durante la creazione del cluster Amazon EKS. Puoi installare l'agente e CloudWatch l'agente SSM utilizzando la `preBootstrapCommands` proprietà dallo schema del file di `eksctl` configurazione (documentazione Weaveworks). Quindi, puoi utilizzare l'agente SSM per connetterti ai tuoi nodi di lavoro senza utilizzare una coppia di chiavi Amazon Elastic Compute Cloud EC2 (Amazon). Inoltre, puoi utilizzare l' CloudWatch agente per monitorare l'utilizzo della memoria e del disco sui nodi di lavoro Amazon EKS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- L'utilità da riga di comando eksctl, installata e configurata su macOS, Linux o Windows
- L'utilità da riga di comando kubectl, installata e configurata su macOS, Linux o Windows

Limitazioni

- Ti consigliamo di evitare di aggiungere script di lunga durata alla `preBootstrapCommands` proprietà, poiché ciò ritarda l'adesione del nodo al cluster Amazon EKS durante le attività di scalabilità. Ti consigliamo invece di creare un'Amazon Machine Image (AMI) personalizzata.
- Questo modello si applica solo alle istanze Amazon EC2 Linux.

Architecture

Stack tecnologico

- Amazon CloudWatch
- Amazon Elastic Kubernetes Service (Amazon EKS)

- AWS Systems Manager Parameter Store

Architettura Target

Il diagramma seguente mostra un esempio di utente che si connette ai nodi di lavoro di Amazon EKS utilizzando l'agente SSM che è stato installato utilizzando. `preBootstrapCommands`

Il diagramma mostra il flusso di lavoro seguente:

1. L'utente crea un cluster Amazon EKS utilizzando il file di `eksctl` configurazione con la `preBootstrapCommands` proprietà, che installa l'agente e CloudWatch l'agente SSM.
2. Tutte le nuove istanze che si uniscono al cluster in un secondo momento a causa delle attività di scalabilità vengono create con l'agente e l'agente SSM preinstallati. CloudWatch
3. L'utente si connette ad Amazon EC2 utilizzando l'agente SSM e quindi monitora l'utilizzo della memoria e del disco utilizzando l'agente. CloudWatch

Tools (Strumenti)

- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue risorse AWS e delle applicazioni che esegui su AWS in tempo reale.
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire Kubernetes su AWS senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- AWS Systems Manager Parameter Store fornisce uno storage sicuro e gerarchico per la gestione dei dati di configurazione e la gestione dei segreti.
- AWS Systems Manager Session Manager ti aiuta a gestire le EC2 istanze, le istanze locali e le macchine virtuali tramite una shell interattiva basata su browser con un solo clic o tramite l'AWS Command Line Interface (AWS CLI).
- eksctl è un'utilità da riga di comando per la creazione e la gestione di cluster Kubernetes su Amazon EKS.
- kubectl è un'utilità a riga di comando per comunicare con il server API del cluster.

Poemi epici

Crea un cluster Amazon EKS.

Operazione	Description	Competenze richieste
Memorizza il file di configura zione CloudWatch dell'agente.	Archivia il file di configura zione dell' CloudWatch agente in AWS Systems Manager Parameter Store nella regione AWS in cui desideri creare il tuo cluster Amazon EKS. A tale scopo, crea un parametro in AWS Systems Manager Parameter Store e annota il nome del parametro (ad esempio,`AmazonClo udwatch-linux`). Per ulteriori informazioni, consulta il codice del file di configurazione dell' CloudWatc h agente di esempio nella sezione Informazioni aggiuntiv e di questo modello.	DevOps ingegnere
Crea il file di configurazione eksctl e il cluster.	1. Crea un file `eksctl` di configurazione che includa i passaggi di installaz ione dell' CloudWatch agente e dell'agente SSM. Per ulteriori informazi oni, consulta l'esempio di codice del file di configura zione eksctl nella sezione Informazioni aggiuntive di questo modello.	AWS DevOps

Operazione	Description	Competenze richieste
	2. Crea un cluster eseguendo il `eksctl create cluster -f cluster.yaml` comando.	

Verifica che l'agente SSM e l' CloudWatch agente funzionino

Operazione	Description	Competenze richieste
Prova l'agente SSM.	Usa SSH per connetterti ai nodi del cluster Amazon EKS utilizzando uno dei metodi descritti in Avvia una sessione dalla documentazione di AWS Systems Manager.	AWS DevOps
Testa l' CloudWatch agente.	Usa la CloudWatch console per convalidare l' CloudWatch agente: 1. Accedi alla console di gestione AWS e apri la console CloudWatch . 2. Nel riquadro di navigazione, espandi Metriche, quindi scegli Tutte le metriche. 3. Nella casella di ricerca della scheda Sfoglia, inserisci e scegli le CWAgent metriche per visualizzare le metriche della memoria e del disco.	AWS DevOps

Risorse correlate

- Installazione ed esecuzione dell' CloudWatch agente sui tuoi server (CloudWatch documentazione Amazon)
- Creare un parametro Systems Manager (console) (documentazione di AWS Systems Manager)
- Crea il file di configurazione CloudWatch dell'agente (CloudWatch documentazione Amazon)
- Avvio di una sessione (AWS CLI) (documentazione di AWS Systems Manager)
- Avvio di una sessione (EC2 console Amazon) (documentazione AWS Systems Manager)

Informazioni aggiuntive

Esempio di file di configurazione CloudWatch dell'agente

Nell'esempio seguente, l' CloudWatch agente è configurato per monitorare l'utilizzo del disco e della memoria su istanze Amazon Linux:

```
{
    "agent": {
        "metrics_collection_interval": 60,
        "run_as_user": "cwagent"
    },
    "metrics": {
        "append_dimensions": {
            "AutoScalingGroupName": "${aws:AutoScalingGroupName}",
            "ImageId": "${aws:ImageId}",
            "InstanceId": "${aws:InstanceId}",
            "InstanceType": "${aws:InstanceType}"
        },
        "metrics_collected": {
            "disk": {
                "measurement": [
                    "used_percent"
                ],
                "metrics_collection_interval": 60,
                "resources": [
                    "*"
                ]
            },
            "mem": {
                "measurement": [
```

```
                    "mem_used_percent"
                ],
                "metrics_collection_interval": 60
            }
        }
    }
}
```

Esempio di file di configurazione eksctl

```
apiVersion: eksctl.io/v1alpha5
kind: ClusterConfig
metadata:
  name: test
  region: us-east-2
  version: "1.24"
managedNodeGroups:
  - name: test
    minSize: 2
    maxSize: 4
    desiredCapacity: 2
    volumeSize: 20
    instanceType: t3.medium
    preBootstrapCommands:
    - sudo yum install amazon-ssm-agent -y
    - sudo systemctl enable amazon-ssm-agent
    - sudo systemctl start amazon-ssm-agent
    - sudo yum install amazon-cloudwatch-agent -y
    - sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-
config -m ec2 -s -c ssm:AmazonCloudwatch-linux
    iam:
      attachPolicyARNs:
        - arn:aws:iam::aws:policy/AmazonEKSWorkerNodePolicy
        - arn:aws:iam::aws:policy/AmazonEKS_CNI_Policy
        - arn:aws:iam::aws:policy/AmazonEC2ContainerRegistryReadOnly
        - arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy
        - arn:aws:iam::aws:policy/AmazonSSMManagedInstanceCore
```

Dettagli aggiuntivi sul codice

- Nell'ultima riga della `preBootstrapCommands` proprietà, `AmazonCloudwatch-linux` c'è il nome del parametro creato in AWS System Manager Parameter Store. Devi includerlo `AmazonCloudwatch-linux` in Parameter Store nella stessa regione AWS in cui hai creato il

cluster Amazon EKS. È anche possibile specificare un percorso di file, ma si consiglia di utilizzare Systems Manager per semplificare l'automazione e la riutilizzabilità.

- Se lo usi `preBootstrapCommands` nel file di `eksctl` configurazione, vedrai due modelli di avvio nella Console di gestione AWS. Il primo modello di avvio include i comandi specificati in`preBootstrapCommands`. Il secondo modello include i comandi specificati `preBootstrapCommands` e i dati utente predefiniti di Amazon EKS. Questi dati sono necessari per far sì che i nodi entrino a far parte del cluster. Il gruppo Auto Scaling del gruppo di nodi utilizza questi dati utente per avviare nuove istanze.
- Se utilizzi l'`iam`attributo nel file di `eksctl` configurazione, devi elencare le policy Amazon EKS predefinite con eventuali policy aggiuntive richieste nelle policy AWS Identity and Access Management (IAM) allegate. Nel frammento di codice del passaggio Crea il file di configurazione eksctl e il cluster, `AmazonSSMMangedInstanceCore` sono state aggiunte politiche aggiuntive per garantire che l' CloudWatch agente `CloudWatchAgentServerPolicy` e l'agente SSM funzionino come previsto. Le `AmazonEC2ContainerRegistryReadOnly` politiche `AmazonEKSWorkerNodePolicyAmazonEKS_CNI_Policy`,, sono politiche obbligatorie necessarie per il corretto funzionamento del cluster Amazon EKS.

Esegui la migrazione dei controller di ingresso NGINX quando abiliti la modalità automatica di Amazon EKS

Olawale Olaleye e Shamanth Devagari, Amazon Web Services

Riepilogo

La modalità automatica EKS per Amazon Elastic Kubernetes Service (Amazon EKS) può ridurre il sovraccarico operativo legato all'esecuzione dei carichi di lavoro sui cluster Kubernetes. Questa modalità consente anche di configurare e gestire AWS l'infrastruttura per tuo conto. Quando abiliti la modalità automatica EKS su un cluster esistente, devi pianificare attentamente la migrazione delle configurazioni di NGINX Ingress Controller. Questo perché il trasferimento diretto dei Network Load Balancer non è possibile.

Puoi utilizzare una strategia di blue/green distribuzione per migrare un'istanza di NGINX Ingress Controller quando abiliti EKS Auto Mode in un cluster Amazon EKS esistente.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un cluster Amazon EKS che esegue Kubernetes versione 1.29 o successiva
- Componenti aggiuntivi Amazon EKS con versioni minime
- Ultima versione di kubectl
- Un'istanza esistente di NGINX Ingress Controller
- (Facoltativo) Una zona ospitata in Amazon Route 53 per lo spostamento del traffico basato su DNS

Architecture

Una distribuzione blu/verde è una strategia di distribuzione in cui si creano due ambienti separati ma identici. Blue/green le implementazioni offrono funzionalità di downtime, release e rollback quasi nulle. L'idea fondamentale è spostare il traffico tra due ambienti identici che eseguono versioni diverse dell'applicazione.

L'immagine seguente mostra la migrazione dei Network Load Balancer da due diverse istanze di NGINX Ingress Controller quando si abilita la modalità automatica EKS. Si utilizza una blue/green distribuzione per spostare il traffico tra i due Network Load Balancer.

Lo spazio dei nomi originale è lo spazio dei nomi blu. È qui che vengono eseguiti il servizio e l'istanza originali di NGINX Ingress Controller, prima di abilitare la modalità automatica EKS. Il servizio e l'istanza originali si connettono a un Network Load Balancer con un nome DNS configurato in Route 53. Il AWS Load Balancer Controller ha distribuito questo Network Load Balancer nel cloud privato virtuale (VPC) di destinazione.

Il diagramma mostra il seguente flusso di lavoro per configurare un ambiente per una distribuzione: blue/green

1. Installa e configura un'altra istanza di NGINX Ingress Controller in un namespace diverso, uno spazio dei nomi verde.
2. In Route 53, configura un nome DNS per un nuovo Network Load Balancer.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire AWS Kubernetes senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- Elastic Load Balancing distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni. Ad esempio, puoi distribuire il traffico tra istanze Amazon Elastic Compute Cloud (Amazon EC2), contenitori e indirizzi IP in una o più zone di disponibilità.
- Amazon Route 53 è un servizio Web DNS altamente scalabile e disponibile.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.

Altri strumenti

- Helm è un gestore di pacchetti open source per Kubernetes che ti aiuta a installare e gestire le applicazioni sul tuo cluster Kubernetes.
- kubectl è un'interfaccia a riga di comando che consente di eseguire comandi sui cluster Kubernetes.
- NGINX Ingress Controller collega le app e i servizi Kubernetes con la gestione delle richieste, l'autenticazione, le risorse personalizzate self-service e il debug.

Epiche

Esamina l'ambiente esistente

Operazione	Description	Competenze richieste

Verifica che l'istanza originale di NGINX Ingress Controller sia operativa.	Immettete il seguente comando per verificare che le risorse nel `ingress-nginx` namespace siano operative . Se hai distribuito NGINX Ingress Controller in un altro spazio dei nomi, aggiorna il nome dello spazio dei nomi in questo comando.
	`kubectl get all -n ingress-nginx`
	Nell'output, conferma che i pod di NGINX Ingress Controller siano in esecuzione. Di seguito è riportato un esempio di output:
	`NAME READY STATUS RESTARTS AGE pod/ingress-nginx-admission-create-xqn9d 0/1 Completed 0 88m pod/ingress-nginx-admission-patch-lhk4j 0/1 Completed 1 88m pod/ingress-nginx-controller-68f68f859-`
		DevOps ingegnere

Operazione	Description	Competenze richieste
	xrz74 1/1 Running 2 (10m ago) 72m NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE service/ingress-nginx-controller LoadBalancer 10.100.67.255 k8s-ingressn-ingressn-abcdefg-12345.elb.eu-west-1.amazonaws.com 80:30330/TCP,443:31462/TCP 88m service/ingress-nginx-controller-admission ClusterIP 10.100.201.176 <none> 443/TCP 88m NAME READY UP-TO-DATE AVAILABLE AGE deployment.apps/ingress-nginx-controller 1/1 1 1 88m NAME	

Operazione	Description	Competenze richieste
	`DESIRED CURRENT READY AGE replicaset.apps/ingress-nginx-controller-68f68f859 1 1 1 72m replicaset.apps/ingress-nginx-controller-d8c96cf68 0 0 0 88m NAME STATUS COMPLETIONS DURATION AGE job.batch/ingress-nginx-admission-create Complete 1/1 4s 88m job.batch/ingress-nginx-admission-patch Complete 1/1 5s 88m`	

Implementa un HTTPd carico di lavoro di esempio per utilizzare NGINX Ingress Controller

Operazione	Description	Competenze richieste
Crea le risorse Kubernetes.	Inserisci i seguenti comandi per creare un esempio di implementazione, servizio e ingresso Kubernetes: `kubectl create deployment demo --image=httpd --port=80`	DevOps ingegnere

Operazione	Description	Competenze richieste

	`kubectl expose deployment demo`
	
	`kubectl create ingress demo --class=nginx \ --rule nginxautomode.local.dev/=demo:80`	

Operazione	Description	Competenze richieste

Esamina le risorse distribuite.	Immettete il comando seguente per visualizzare un elenco delle risorse distribuite:
	`kubectl get all,ingress`
	Nell'output, confermate che il HTTPd pod di esempio sia in esecuzione. Di seguito è riportato un esempio di output:
	`NAME READY STATUS RESTARTS AGE pod/demo-7d94f8cb4f-q68wc 1/1 Running 0 59m`
	`NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE service/demo ClusterIP 10.100.78.155 <none> 80/TCP 59m service/kubernetes ClusterIP 10.100.0.1 <none> 443/TCP 117m`
	`NAME READY UP-TO-DATE AVAILABLE AGE deployment.apps/demo 1/1 1 1 59m`
		DevOps ingegnere

Operazione	Description	Competenze richieste
	NAME DESIRED CURRENT READY AGE replicaset.apps/demo-7d94f8cb4f 1 1 1 59m NAME CLASS HOSTS ADDRESS PORTS AGE ingress.networking.k8s.io/demo nginx nginxautomode.local.dev k8s-ingressn-ingressn-abcdefg-12345.elb.eu-west-1.amazonaws.com 80 56m	

Operazione	Description	Competenze richieste

Conferma che il servizio è raggiungibile.	Immettere il seguente comando per confermare che il servizio è raggiungibile tramite il nome DNS del Network Load Balancer:
	`curl -H "Host: nginxautomode.local.dev" http://k8s-ingressn-ingressn-abcdefg-12345.elb.eu-west-1.amazonaws.com`
	L'output previsto è il seguente:
	`<html><body><h1>It works!</h1></body></html>`
		DevOps ingegnere

Operazione	Description	Competenze richieste

(Facoltativo) Crea un record DNS.	1. Segui le istruzioni in Creazione di record utilizzando la console Amazon Route 53 (documentazione Route 53) per creare un record DNS per il dominio configurato.
	2. Inserisci il seguente comando per confermare che il servizio è raggiungi bile tramite il nome di dominio configurato:
	`curl "http://nginxautomode.local.dev/?[1-5]"`
	L'output previsto è il seguente:
	`<html><body><h1>It works!</h1></body></html>`
	`<html><body><h1>It works!</h1></body></html>`
	`<html><body><h1>It works!</h1></body></html>`
	`<html><body><h1>It works!</h1></body></html>`
	`<html><body><h1>It works!</h1></body></html>`
		DevOps ingegnere, AWS DevOps

Abilita la modalità EKS Auto sul cluster esistente

Operazione	Description	Competenze richieste
Abilita la modalità automatica EKS.	Segui le istruzioni in Abilita la modalità automatica EKS su un cluster esistente (documentazione Amazon EKS).	AWS DevOps

Installa un nuovo controller di ingresso NGINX

Operazione	Description	Competenze richieste
Configura una nuova istanza di NGINX Ingress Controller.	1. Scarica il modello deploy.yaml. 2. Apri il modello deploy.yaml nel tuo editor preferito. 3. Nella `kind: Namespace` sezione, inserisci un nome univoco per lo spazio dei nomi, ad esempio: `ingress-nginx-v2` `apiVersion: v1 kind: Namespace metadata: labels: app.kuber netes.io/instance: ingress-nginx app.kuber netes.io/name: ingress-nginx name: ingress-n ginx-v2`	DevOps ingegnere

Operazione	Description	Competenze richieste

	4. Per ogni sezione, aggiorna il `namespace` valore con il nuovo nome.
	5. Nella `kind: Deployment` sezione, effettuate le seguenti operazioni:
	a. Immettete un valore univoco per`--controller-class` , ad esempio`k8s.io/ingress-nginx-v2` .
	b. Inserisci un valore univoco per`--ingress-class` , ad esempio`nginx-v2`.
	`apiVersion: apps/v1`
	`kind: Deployment`
	`  name: ingress-nginx-controller`
	`  namespace: ingress-nginx-v2`
	`...`
	`    spec:`
	`      containers:`
	`      - args:`
	`        - /nginx-ingress-controller`
	`        - --publish-service=$(POD_NAMESPACE)/ingress-nginx-controller`
	`        - --election-id=ingress-nginx-leader`
	`        - --controller-class=k8s.io/ingress-nginx-v2`	

Operazione	Description	Competenze richieste

	<pre>- --ingress-class=nginx-v2</pre> 6. Nella `kind: IngressClass` sezione, inserisci gli stessi valori per `--controller-class` e `--ingress-class` che hai usato nella sezione precedente: <pre>apiVersion: networking.k8s.io/v1
	kind: IngressClass
	metadata:
	  labels:
	  app.kubernetes.io/component: controller
	  app.kubernetes.io/instance: ingress-nginx
	  app.kubernetes.io/name: ingress-nginx
	  app.kubernetes.io/part-of: ingress-nginx
	  app.kubernetes.io/version: 1.12.0
	  name: nginx-v2
	spec:
	  controller: k8s.io/ingress-nginx-v2</pre> 7. Nella sezione seguente, aggiungi `loadBalancerClass: eks.amazonaws.com/nlb` per	

Operazione	Description	Competenze richieste

	effettuare il provisioning di un Network Load Balancer per l'istanza di NGINX Ingress Controller:
	
	`apiVersion: v1`
	`kind: Service`
	`metadata:`
	`  name: ingress-nginx-controller`
	`  namespace: ingress-nginx-v2`
	`spec:`
	`...`
	`  selector:`
	`  app.kubernetes.io/component: controller`
	`  app.kubernetes.io/instance: ingress-nginx`
	`  app.kubernetes.io/name: ingress-nginx`
	`  type: LoadBalancer`
	`  loadBalancerClass: eks.amazonaws.com/nlb`
	
	8. Salva e chiudi il modello deploy.yaml.	

Distribuisci la nuova istanza di NGINX Instance Controller.	Immettete il seguente comando per applicare il file manifest modificato: `kubectl apply -f deploy.yaml`	DevOps ingegnere

Operazione	Description	Competenze richieste

Conferma la corretta implementazione.	Immettete il comando seguente per verificare che le risorse nel `ingress-nginx-v2` namespace siano operative:
	`kubectl get all -n ingress-nginx-v2`
	Nell'output, conferma che i pod di NGINX Ingress Controller siano in esecuzione. Di seguito è riportato un esempio di output:
	`NAME READY STATUS RESTARTS AGE pod/ingress-nginx-admission-create-7shrj 0/1 Completed 0 24s pod/ingress-nginx-admission-patch-vkxr5 0/1 Completed 1 24s pod/ingress-nginx-controller-757bfcbc6d-4fw52 1/1 Running 0 24s NAME TYPE CLUSTER-IP EXTERNAL-IP`
		DevOps ingegnere

Operazione	Description	Competenze richieste
	PORT(S) AGE service/ingress- nginx-controller LoadBalancer 10.100.208.114 k8s-ingressn-ingre ssn-2e5e37fab6-848 337cd9c9d520f.elb.eu- west-1.amazonaws.com 80:31469/TCP,443:3 0658/TCP 24s service/ingress- nginx-controller- admission ClusterIP 10.100.150.114 <none> 443/TCP 24s NAME READY UP-TO-DATE AVAILABLE AGE deployment.apps/ ingress-nginx-co ntroller 1/1 1 1 24s NAME DESIRED CURRENT READY AGE replicaset.apps/ ingress-nginx-co ntroller-757bfcbc6d	

Operazione	Description	Competenze richieste
	`1 1 1 24s NAME STATUS COMPLETIONS DURATION AGE job.batch/ingress-nginx-admission-create Complete 1/1 4s 24s job.batch/ingress-nginx-admission-patch Complete 1/1 5s 24s`	
Crea un nuovo ingresso per il carico di HTTPd lavoro di esempio.	Immettete il seguente comando per creare un nuovo ingresso per il carico di lavoro di esempio esistente: HTTPd `kubectl create ingress demo-new --class=nginx-v2 \ --rule nginxautomode.local.dev/=demo:80`	DevOps ingegnere

Operazione	Description	Competenze richieste

Conferma che il nuovo ingresso funzioni.	Immettete il seguente comando per confermare che il nuovo ingresso funzioni:
	`curl -H "Host: nginxautomode.local.dev" k8s-ingressn-ingressn-2e5e37fab6-848337cd9c9d520f.elb.eu-west-1.amazonaws.com`
	L'output previsto è il seguente:
	`<html><body><h1>It works!</h1></body></html>`
		DevOps ingegnere

tagliare

Operazione	Description	Competenze richieste
Passa al nuovo namespace.	1. (Facoltativo) Segui le istruzioni in Modifica dei record (documentazione di Route 53) per aggiornare il record DNS. 2. Dopo aver confermato che la nuova istanza di NGINX Ingress Controller funziona come previsto, elimina l'originale. 3. Eliminare il Load Balancer AWS Controller autogesti to. Per istruzioni, consulta	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	Migrare app da ALB Ingress Controller obsoleto (documentazione Amazon EKS). 4. Svuota i gruppi di nodi gestiti. Per istruzioni, vedere Eliminazione e drenaggio dei gruppi di nodi (documentazione eksctl).	

Operazione	Description	Competenze richieste

Controlla i due ingressi.	Immettete il comando seguente per esaminare i due ingressi creati per il carico di lavoro di esempio HTTPd :
	`kubectl get ingress`
	Di seguito è riportato un esempio di output:
	`NAME CLASS HOSTS ADDRESS PORTS AGE`
	`demo nginx nginxautomode.local.dev k8s-ingressn-ingressn-abcdefg-12345.elb.eu-west-1.amazonaws.com 80 95m`
	`demo-new nginx-v2 nginxautomode.local.dev k8s-ingressn-ingressn-2e5e37fab6-848337cd9c9d520f.elb.eu-west-1.amazonaws.com 80 33s`
		DevOps ingegnere

Risorse correlate

- Abilita la modalità automatica EKS su un cluster esistente (documentazione Amazon EKS)

- Risolvi i problemi relativi ai sistemi di bilanciamento del carico creati dal controller di servizio Kubernetes in Amazon EKS (re:POST Knowledge Center)AWS
- NGINX Ingress Controller (documentazione NGINX)

Esegui la migrazione dei carichi di lavoro dei container da Azure Red Hat OpenShift (ARO) a Servizio Red Hat OpenShift su AWS (ROSA)

Naveen Ramasamy, Srikanth Rangavajhala e Gireesh Sreekantan, Amazon Web Services

Riepilogo

Questo modello fornisce step-by-step istruzioni per la migrazione dei carichi di lavoro dei container da Azure Red Hat (ARO) a (ROSA). OpenShift Servizio Red Hat OpenShift su AWS ROSA è un servizio Kubernetes gestito fornito da Red Hat in collaborazione con. AWS Ti aiuta a implementare, gestire e scalare le tue applicazioni containerizzate utilizzando la piattaforma Kubernetes e sfrutta sia l'esperienza di Red Hat in Kubernetes che l'infrastruttura. Cloud AWS

La migrazione dei carichi di lavoro dei container da ARO, da altri cloud o dall'ambiente locale a ROSA comporta il trasferimento di applicazioni, configurazioni e dati da una piattaforma all'altra. Questo modello aiuta a garantire una transizione fluida ottimizzando al contempo Cloud AWS i servizi, la sicurezza e l'efficienza dei costi. Copre due metodi per la migrazione dei carichi di lavoro verso i cluster ROSA: CI/CD e Migration Toolkit for Containers (MTC).

Questo modello copre entrambi i metodi. Il metodo scelto dipende dalla complessità e dalla certezza del processo di migrazione. Se hai il pieno controllo sullo stato dell'applicazione e puoi garantire una configurazione coerente attraverso una pipeline, ti consigliamo di utilizzare questo CI/CD metodo. Tuttavia, se lo stato dell'applicazione comporta incertezze, cambiamenti imprevisti o un ecosistema complesso, consigliamo di utilizzare MTC come percorso affidabile e controllato per migrare l'applicazione e i relativi dati in un nuovo cluster. Per un confronto dettagliato dei due metodi, consultate la sezione Informazioni aggiuntive.

Vantaggi della migrazione a ROSA:

- ROSA si integra perfettamente con AWS come servizio nativo. È facilmente accessibile tramite Console di gestione AWS e fatturato tramite un unico. Account AWS Offre la piena compatibilità con gli altri Servizi AWS e fornisce supporto collaborativo sia da parte di Red Hat che da parte di Red AWS Hat.
- ROSA supporta implementazioni ibride e multi-cloud. Consente alle applicazioni di funzionare in modo coerente tra data center locali e più ambienti cloud.
- ROSA trae vantaggio dall'attenzione rivolta da Red Hat alla sicurezza e offre funzionalità come il controllo degli accessi basato sui ruoli (RBAC), la scansione delle immagini e la valutazione delle vulnerabilità per garantire un ambiente container sicuro.

- ROSA è progettato per scalare facilmente le applicazioni e offre opzioni di alta disponibilità. Consente alle applicazioni di crescere secondo necessità mantenendo l'affidabilità.
- ROSA automatizza e semplifica l'implementazione di un cluster Kubernetes rispetto ai metodi di configurazione e gestione manuali. Questo accelera il processo di sviluppo e implementazione.
- ROSA trae vantaggio dai Cloud AWS servizi e offre una perfetta integrazione con AWS offerte come servizi di database, soluzioni di archiviazione e servizi di sicurezza.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Autorizzazioni configurate in base alle quali ROSA si basa per fornire funzionalità. Servizi AWS Per ulteriori informazioni, consulta Prerequisiti nella documentazione ROSA.
- ROSA è abilitato sulla console ROSA. Per istruzioni, consultate la documentazione di ROSA.
- Il cluster ROSA è stato installato e configurato. Per ulteriori informazioni, consulta la Guida introduttiva a ROSA nella documentazione di ROSA. Per comprendere i diversi metodi di configurazione di un cluster ROSA, consulta la Guida AWS prescrittiva alle strategie di implementazione di ROSA.
- Connettività di rete stabilita dalla rete locale a quella AWS passante AWS Direct Connect(preferita) o AWS Virtual Private Network ().Site-to-Site VPN
- Un'istanza Amazon Elastic Compute Cloud (Amazon EC2) o un altro server virtuale per installare strumenti come `aws client` client OpenShift CLI `oc` (), client ROSA e binario Git.

Prerequisiti aggiuntivi per il metodo: CI/CD

- Accesso al server Jenkins locale con le autorizzazioni per creare una nuova pipeline, aggiungere fasi, aggiungere OpenShift cluster ed eseguire build.
- Accesso al repository Git in cui viene mantenuto il codice sorgente dell'applicazione, con le autorizzazioni per creare un nuovo ramo Git ed eseguire commit sul nuovo ramo.

Prerequisiti aggiuntivi per il metodo MTC:

- Un bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3), che verrà utilizzato come repository di replica.

- Accesso amministrativo al cluster ARO di origine. Ciò è necessario per configurare la connessione MTC.

Limitazioni

- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Architecture

ROSA offre tre modelli di distribuzione di rete: pubblico, privato e. AWS PrivateLink PrivateLinkconsente ai team di Red Hat Site Reliability Engineering (SRE) di gestire il cluster utilizzando una sottorete privata connessa all' PrivateLink endpoint del cluster in un VPC esistente.

La scelta dell' PrivateLinkopzione offre la configurazione più sicura. Per questo motivo, la consigliamo per carichi di lavoro sensibili o requisiti di conformità rigorosi. Per informazioni sulle opzioni di implementazione di reti pubbliche e private, consulta la OpenShift documentazione di Red Hat.

> **Important**
>
> È possibile creare un PrivateLink cluster solo al momento dell'installazione. Non è possibile modificare un cluster da utilizzare PrivateLink dopo l'installazione.

Il diagramma seguente illustra l' PrivateLink architettura di un cluster ROSA utilizzato Direct Connect per connettersi agli ambienti locali e ARO.

AWS autorizzazioni per ROSA

Per AWS le autorizzazioni a ROSA, ti consigliamo di utilizzare AWS Security Token Service (AWS STS) con token dinamici di breve durata. Questo metodo utilizza ruoli e politiche predefiniti con privilegi minimi per concedere a ROSA le autorizzazioni minime per operare in e supporta l'installazione Account AWS, il piano di controllo e le funzionalità di calcolo di ROSA.

Ridistribuzione della pipeline CI/CD

CI/CD pipeline redeployment is the recommended method for users who have a mature CI/CDgasdotto. Scegliendo questa opzione, è possibile utilizzare qualsiasi strategia di DevOps implementazione per spostare gradualmente il carico delle applicazioni verso le distribuzioni su ROSA.

> **Note**
>
> Questo modello presuppone un caso d'uso comune in cui si dispone di una pipeline Git, JFrog Artifactory e Jenkins locali. Questo approccio richiede di stabilire la connettività di rete dalla rete locale a AWS quella esterna e di configurare il cluster ROSA prima di seguire le istruzioni nella sezione Epics. Direct Connect Per i dettagli, consulta la sezione Prerequisiti.

Il diagramma seguente mostra il flusso di lavoro per questo metodo.

Metodo MTC

Puoi utilizzare Migration Toolkit for Containers (MTC) per migrare i carichi di lavoro containerizzati tra diversi ambienti Kubernetes, ad esempio da ARO a ROSA. MTC semplifica il processo di migrazione automatizzando diverse attività chiave e fornendo un framework completo per la gestione del ciclo di vita della migrazione.

Il diagramma seguente mostra il flusso di lavoro per questo metodo.

Tools (Strumenti)

Servizi AWS

- AWS DataSyncè un servizio di trasferimento e individuazione di dati online che consente di spostare file o dati di oggetti da, verso e tra i servizi di AWS archiviazione.
- AWS Direct Connectcollega la rete interna a una Direct Connect posizione tramite un cavo Ethernet standard in fibra ottica. Con questa connessione, è possibile creare interfacce virtuali direttamente al pubblico Servizi AWS ignorando i provider di servizi Internet nel percorso di rete.
- AWS PrivateLinkti aiuta a creare connessioni private unidirezionali dai tuoi cloud privati virtuali (VPCs) a servizi esterni al VPC.
- Servizio Red Hat OpenShift su AWS (ROSA) è un servizio gestito che aiuta OpenShift gli utenti Red Hat a creare, scalare e gestire applicazioni containerizzate su. AWS

- AWS Security Token Service (AWS STS) consente di richiedere credenziali temporanee con privilegi limitati per gli utenti.

Altri strumenti

- Migration Toolkit for Containers (MTC) fornisce una console e un'API per la migrazione di applicazioni containerizzate da ARO a ROSA.

Best practice

- Per la resilienza e se disponi di carichi di lavoro di conformità alla sicurezza, configura un cluster ROSA Multi-AZ che utilizzi. PrivateLink Per ulteriori informazioni, consulta la documentazione ROSA.

> **Note**
>
> PrivateLink non può essere configurato dopo l'installazione.

- Il bucket S3 utilizzato per l'archivio di replica non deve essere pubblico. Utilizza le policy dei bucket S3 appropriate per limitare l'accesso.
- Se scegli il metodo MTC, utilizza l'opzione di migrazione Stage per ridurre la finestra di inattività durante il cutover.
- Controllate le quote di servizio prima e dopo il provisioning del cluster ROSA. Se necessario, richiedete un aumento della quota in base alle vostre esigenze. Per ulteriori informazioni, consulta la pagina relativa alla Documentazione sulle quote di servizio.
- Rivedi le linee guida sulla sicurezza ROSA e implementa le migliori pratiche di sicurezza.
- Si consiglia di rimuovere l'amministratore predefinito del cluster dopo l'installazione. Per ulteriori informazioni, consulta la OpenShift documentazione di Red Hat.
- Utilizza la scalabilità automatica del pool di macchine per ridurre i nodi di lavoro inutilizzati nel cluster ROSA e ottimizzare i costi. Per ulteriori informazioni, consultate il ROSA Workshop.
- Utilizza il servizio Red Hat Cost Management for OpenShift Container Platform per comprendere e monitorare meglio i costi di cloud e container. Per ulteriori informazioni, consulta il ROSA Workshop.
- Monitora e verifica i servizi e le applicazioni dell'infrastruttura del cluster ROSA utilizzando Servizi AWS. Per ulteriori informazioni, vedere il ROSA Workshop.

Epiche

Opzione 1: utilizzare una pipeline CI/CD

Operazione	Description	Competenze richieste
Aggiungi il nuovo cluster ROSA a Jenkins.	1. Sulla console Jenkins, scegli Gestisci Jenkins, Configura sistema. 2. Nella pagina Gestisci Jenkins, nella sezione OpenShift Plug-in, scegli Aggiungi un nuovo cluster. 3. Fornisci le informazioni richieste come il nome del cluster, l'URL del server API e le informazioni sul token per l'autenticazione con ROSA.	Amministratore AWS, amministratore di sistema AWS, AWS DevOps
Aggiungi il `oc` client ai tuoi nodi Jenkins.	1. Sulla console Jenkins, scegli Manage Jenkins, Global Tool Configuration. 2. Nella sezione OpenShift Client Tools, installa la versione del client OpenShift CLI (`oc`) che è la stessa della versione del cluster ROSA.	Amministratore AWS, amministratore di sistema AWS, AWS DevOps
Crea un nuovo ramo Git.	Crea un nuovo ramo nel tuo repository Git per`rosa-dev`. Questo ramo separa il codice o le modifiche ai parametri di configurazione per ROSA dalla pipeline esistente.	AWS DevOps

Operazione	Description	Competenze richieste
Tagga le immagini per ROSA.	Nella fase di creazione, usa un tag diverso per identificare le immagini create dalla pipeline ROSA.	Amministratore AWS, amministratore di sistema AWS, AWS DevOps
Creare una pipeline.	Crea una nuova pipeline Jenkins simile alla tua pipeline esistente. Per questa pipeline, usa il ramo `rosa-dev` Git che hai creato in precedenz a e assicurati di includere le fasi di checkout, scansione del codice e compilazione di Git identiche alla pipeline esistente.	Amministratore AWS, amministratore di sistema AWS, AWS DevOps
Aggiungi una fase di distribuz ione ROSA.	Nella nuova pipeline, aggiungi una fase da implementare nel cluster ROSA e fai riferimen to al cluster ROSA che hai aggiunto alla configurazione globale di Jenkins.	Amministratore AWS, AWS DevOps, amministratore di sistema AWS
Inizia una nuova build.	In Jenkins, seleziona la tua pipeline e scegli Costruisci ora, oppure inizia una nuova build eseguendo una modifica al ramo `rosa-dev` in Git.	Amministratore AWS, AWS DevOps, amministratore di sistema AWS
Verifica la distribuzione.	Utilizzate il comando oc o la console ROSA per verificar e che l'applicazione sia stata distribuita sul cluster ROSA di destinazione.	Amministratore AWS, AWS DevOps, amministratore di sistema AWS

Operazione	Description	Competenze richieste
Copia i dati nel cluster di destinazione.	Per i carichi di lavoro con stato, copia i dati dal cluster di origine al cluster di destinazione utilizzando AWS DataSync le nostre utilità open source come rsync oppure puoi utilizzare il metodo MTC. Per ulteriori informazioni, consulta la documentazione relativa ad AWS DataSync.	Amministratore AWS, AWS DevOps, amministratore di sistema AWS
Testa la tua applicazione.	1. Recupera l'URL del percorso per la tua applicazione utilizzando il comando oc route o la console ROSA. 2. Testa la tua applicazione utilizzando l'URL del percorso.	Amministratore AWS, AWS DevOps, amministratore di sistema AWS
Tagliare.	Se il test ha esito positivo, utilizza la policy Amazon Route 53 appropriata per spostare il traffico dall'applicazione ospitata da ARO all'applicazione ospitata da ROSA. Una volta completato questo passaggio, il carico di lavoro dell'applicazione passerà completamente al cluster ROSA.	Amministratore AWS, amministratore di sistema AWS

Opzione 2: usa MTC

Operazione	Description	Competenze richieste
Installa l'operatore MTC.	Installa l'operatore MTC su entrambi i cluster ARO e ROSA: 1. Nella console ARO o ROSA, vai alla pagina Operatori. OperatorHub 2. Nella casella Filtra per parola chiave, trova o digita MTC. 3. Selezionate l'operatore MTC per visualizzare informazioni aggiuntive. 4. Dopo aver letto le informazi oni sull'operatore, scegliete Installa.	Amministratore AWS, AWS DevOps, amministratore di sistema AWS
Configura il traffico di rete verso l'archivio di replica.	Se utilizzi un server proxy, configuralo per consentire il traffico di rete tra l'archivio di replica e i cluster. Il repositor y di replica è un oggetto di storage intermedio che MTC utilizza per migrare i dati. I cluster di origine e di destinazi one devono avere accesso di rete al repository di replica durante la migrazione.	Amministratore AWS, AWS DevOps, amministratore di sistema AWS
Aggiungi il cluster di origine a MTC.	Sulla console web MTC, aggiungete il cluster di origine ARO.	Amministratore AWS, AWS DevOps, amministratore di sistema AWS

Operazione	Description	Competenze richieste
Aggiungi Amazon S3 come repository di replica.	Sulla console web MTC, aggiungi il bucket Amazon S3 (vedi Prerequisiti) come repository di replica.	Amministratore AWS, AWS DevOps, amministratore di sistema AWS
Crea un piano di migrazione.	Sulla console web MTC, create un piano di migrazione e specificate il tipo di trasferimento dei dati come Copia. Ciò indicherà a MTC di copiare i dati dal cluster di origine (ARO) al bucket S3 e dal bucket al cluster di destinazione (ROSA).	Amministratore AWS, AWS DevOps, amministratore di sistema AWS

Operazione	Description	Competenze richieste
Esegui il piano di migrazione.	Esegui il piano di migrazione utilizzando l'opzione Stage o Cutover: • Scegli Stage per copiare i dati nel cluster di destinazione senza interrompere l'applicazione. Puoi eseguire migrazioni in più fasi per copiare la maggior parte dei dati prima della migrazione completa. Ciò riduce la durata della migrazione cutover. • Scegliete Cutover per interrompere l'applicazione sul cluster di origine mentre spostate le risorse nel cluster di destinazione. Per evitare l'arresto dell'applicazione, è possibile deselezionare la casella di controllo Interrompi le transazioni sul cluster di origine durante la migrazione.	Amministratore AWS, AWS DevOps, amministratore di sistema AWS

risoluzione dei problemi

Problema	Soluzione
Problemi di connettività	Quando esegui la migrazione dei carichi di lavoro dei container da ARO a ROSA, potresti

Problema	Soluzione
	riscontrare problemi di connettività che devono essere risolti per garantire una migrazione di successo. Per risolvere questi problemi di connettività (elencati in questa tabella) durante la migrazione, sono fondamentali una pianificazione meticolosa, il coordinamento con i team di rete e sicurezza e test approfonditi. L'implementazione di una strategia di migrazion e graduale e la verifica della connettività in ogni fase contribuiranno a ridurre al minimo le potenziali interruzioni e a garantire una transizione fluida da ARO a ROSA.
Differenze di configurazione di rete	ARO e ROSA potrebbero presentare variazion i nelle configurazioni di rete, come le impostazi oni della rete virtuale (VNet), le sottoreti e le politiche di rete. Per una comunicaz ione corretta tra i servizi, assicuratevi che le impostazioni di rete siano allineate tra le due piattaforme.
Regole del gruppo di sicurezza e del firewall	ROSA e ARO potrebbero avere impostazioni predefinite diverse per il gruppo di sicurezza e il firewall. Assicurati di modificare e aggiornar e queste regole per consentire il traffico necessario a mantenere la connettività tra contenitori e servizi durante la migrazione.
Modifiche all'indirizzo IP e al DNS	Durante la migrazione dei carichi di lavoro, gli indirizzi IP e i nomi DNS potrebbero cambiare. Riconfigura le applicazioni che si basano su nomi DNS statici IPs o specifici.

Problema	Soluzione
Accesso ai servizi esterni	Se l'applicazione dipende da servizi esterni come i database oppure APIs, potrebbe essere necessario aggiornare le impostazioni di connessione per assicurarsi che possano comunicare con i nuovi servizi di ROSA.
Configurazione di Azure Private Link	Se usi Azure Private Link o servizi endpoint privati in ARO, dovrai configurare la funzionalità equivalente in ROSA per garantire la connettiv ità privata tra le risorse.
Site-to-Site VPN o configurazione Direct Connect	Se esistono Direct Connect connessioni Site-to-Site VPN o tra la rete locale e ARO, sarà necessario stabilire connessioni simili con ROSA per una comunicazione ininterrotta con le risorse locali.
Impostazioni di ingresso e bilanciamento del carico	Le configurazioni dei controller di ingresso e dei sistemi di bilanciamento del carico potrebbero differire tra ARO e ROSA. Riconfigurate queste impostazioni per mantenere l'accesso esterno ai vostri servizi.
Gestione di certificati e TLS	Se le tue applicazioni utilizzano certificati SSL o TLS, assicurati che i certificati siano validi e configurati correttamente in ROSA.
Accesso al registro dei contenitori	Se i contenitori sono ospitati in un registro di contenitori esterno, imposta l'autenticazione e le autorizzazioni di accesso appropriate per ROSA.

Problema	Soluzione
Monitoraggio e registrazione dei log	Aggiorna le configurazioni di monitoraggio e registrazione in modo da rispecchiare la nuova infrastruttura su ROSA in modo da poter continuare a monitorare efficacemente lo stato e le prestazioni dei tuoi contenitori.

Risorse correlate

AWSriferimenti

- Che cos'è Servizio Red Hat OpenShift su AWS? (documentazione ROSA)
- Inizia a usare ROSA (documentazione ROSA)
- Servizio Red Hat OpenShift su AWS strategie di implementazione (AWS Prescriptive Guidance)
- Servizio Red Hat OpenShift su AWS Now GA (AWS post sul blog)
- Workshop ROSA
- Domande frequenti su ROSA
- Domande frequenti sul seminario ROSA
- Prezzi ROSA

OpenShift Documentazione Red Hat

- Installazione rapida di un cluster su AWS
- Installazione di un cluster AWS in una rete con restrizioni
- Installazione di un cluster AWS in un VPC esistente
- Installazione di un cluster su un'infrastruttura fornita dall'utente tramite modelli AWS CloudFormation
- Installazione di un cluster AWS in una rete limitata con un'infrastruttura fornita dall'utente
- Installazione di un cluster con personalizzazioni AWS
- Guida introduttiva alla OpenShift CLI

Informazioni aggiuntive

Scelta tra MTC e le opzioni di ridistribuzione della CI/CD pipeline

La migrazione delle applicazioni da un OpenShift cluster a un altro richiede un'attenta considerazione. Idealmente, si vorrebbe una transizione fluida utilizzando una CI/CD pipeline per ridistribuire l'applicazione e gestire la migrazione dei dati di volume persistenti. Tuttavia, in pratica, un'applicazione in esecuzione su un cluster è suscettibile di modifiche impreviste nel tempo. Queste modifiche possono far sì che l'applicazione si discosti gradualmente dallo stato di distribuzione originale. MTC offre una soluzione per scenari in cui il contenuto esatto di un namespace è incerto ed è importante una migrazione senza interruzioni di tutti i componenti dell'applicazione in un nuovo cluster.

Fare la scelta giusta richiede la valutazione dello scenario specifico e la valutazione dei vantaggi di ciascun approccio. In questo modo, potete garantire una migrazione efficace e senza interruzioni, in linea con le vostre esigenze e priorità. Di seguito sono riportate ulteriori linee guida per la scelta tra le due opzioni.

Ridistribuzione della pipeline CI/CD

Il metodo della CI/CD pipeline è l'approccio consigliato se l'applicazione può essere ridistribuita con sicurezza utilizzando una pipeline. Ciò garantisce che la migrazione sia controllata, prevedibile e allineata alle pratiche di distribuzione esistenti. Scegliendo questo metodo, è possibile utilizzare strategie di implementazione blu/green o canary per spostare gradualmente il carico sulle implementazioni su ROSA. In questo scenario, questo modello presuppone che Jenkins stia orchestrando le distribuzioni delle applicazioni dall'ambiente locale.

Vantaggi:

- Non è necessario l'accesso amministrativo al cluster ARO di origine né è necessario distribuire operatori sul cluster di origine o di destinazione.
- Questo approccio consente di modificare gradualmente il traffico utilizzando una DevOps strategia.

Svantaggi:

- È necessario uno sforzo maggiore per testare la funzionalità dell'applicazione.
- Se l'applicazione contiene dati persistenti, sono necessari passaggi aggiuntivi per copiare i dati utilizzando AWS DataSync o altri strumenti.

Migrazione MTC

Nel mondo reale, le applicazioni in esecuzione possono subire modifiche impreviste che le allontanano dalla distribuzione iniziale. Scegliete l'opzione MTC quando non siete sicuri dello stato attuale dell'applicazione nel cluster di origine. Ad esempio, se il vostro ecosistema applicativo comprende vari componenti, configurazioni e volumi di archiviazione dei dati, vi consigliamo di scegliere MTC per garantire una migrazione completa che copra l'applicazione e il suo intero ambiente.

Vantaggi:

- MTC fornisce il backup e il ripristino completi del carico di lavoro.
- Copierà i dati persistenti dall'origine alla destinazione durante la migrazione del carico di lavoro.
- Non richiede l'accesso all'archivio del codice sorgente.

Svantaggi:

- Sono necessari i privilegi amministrativi per installare l'operatore MTC sui cluster di origine e di destinazione.
- Il DevOps team richiede una formazione per utilizzare lo strumento MTC ed eseguire le migrazioni.

Esegui attività Amazon ECS su Amazon WorkSpaces con Amazon ECS Anywhere

Akash Kumar, Amazon Web Services

Riepilogo

Amazon Elastic Container Service (Amazon ECS) Anywhere supporta la distribuzione di attività Amazon ECS in qualsiasi ambiente, inclusa l'infrastruttura gestita di Amazon Web Services (AWS) e l'infrastruttura gestita dai clienti. Puoi farlo utilizzando un piano di controllo completamente gestito da AWS, in esecuzione nel cloud e sempre aggiornato.

Le aziende utilizzano spesso Amazon WorkSpaces per lo sviluppo di applicazioni basate su container. Ciò ha richiesto Amazon Elastic Compute Cloud (Amazon EC2) o AWS Fargate con un cluster Amazon ECS per testare ed eseguire le attività ECS. Ora, utilizzando Amazon ECS Anywhere, puoi aggiungere WorkSpaces Amazon come istanze esterne direttamente a un cluster ECS ed eseguire le tue attività direttamente. Ciò riduce i tempi di sviluppo, poiché puoi testare il tuo contenitore con un cluster ECS localmente su Amazon WorkSpaces. Puoi anche risparmiare sui costi di utilizzo EC2 delle istanze Fargate per testare le tue applicazioni container.

Questo modello mostra come distribuire le attività ECS su Amazon WorkSpaces con Amazon ECS Anywhere. Configura il cluster ECS e utilizza AWS Directory Service Simple AD per avviare il WorkSpaces. Quindi l'attività ECS di esempio avvia NGINX in. WorkSpaces

Prerequisiti e limitazioni

- Un account AWS attivo
- Interfaccia a riga di comando di AWS (CLI AWS)
- Credenziali AWS configurate sulla tua macchina

Architecture

Stack tecnologico Target

- Un cloud privato virtuale (VPC)
- Un cluster Amazon ECS
- Amazon WorkSpaces

- AWS Directory Service con Simple AD

Architettura Target

L'architettura include i seguenti servizi e risorse:

- Un cluster ECS con sottoreti pubbliche e private in un VPC personalizzato
- Simple AD nel VPC per fornire agli utenti l'accesso ad Amazon WorkSpaces
- Amazon ha effettuato il WorkSpaces provisioning nel VPC utilizzando Simple AD
- AWS Systems Manager attivato per aggiungere Amazon WorkSpaces come istanze gestite
- Utilizzando Amazon ECS e AWS Systems Manager Agent (SSM Agent), Amazon si è WorkSpaces aggiunto a Systems Manager e al cluster ECS
- Un esempio di attività ECS da eseguire nel cluster ECS WorkSpaces

Tools (Strumenti)

- AWS Directory Service Simple Active Directory (Simple AD) è una directory gestita autonoma alimentata da un server compatibile con Active Directory Samba 4. Simple AD fornisce un sottoinsieme delle funzionalità offerte da AWS Managed Microsoft AD, inclusa la capacità di gestire gli utenti e di connettersi in modo sicuro alle istanze Amazon EC2 .
- Amazon Elastic Container Service (Amazon ECS) è un servizio rapido e scalabile di gestione dei container che ti aiuta a eseguire, arrestare e gestire container in un cluster.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Systems Manager ti aiuta a gestire le applicazioni e l'infrastruttura in esecuzione nel cloud AWS. Semplifica la gestione delle applicazioni e delle risorse, riduce i tempi di rilevamento e risoluzione dei problemi operativi e ti aiuta a gestire le tue risorse AWS in modo sicuro su larga scala.
- Amazon ti WorkSpaces aiuta a fornire desktop Microsoft Windows o Amazon Linux virtuali basati sul cloud per i tuoi utenti, noti come. WorkSpaces WorkSpaces elimina la necessità di procurarsi e distribuire hardware o installare software complessi.

Epiche

Configura il cluster ECS

Operazione	Description	Competenze richieste
Crea e configura il cluster ECS.	Per creare il cluster ECS, segui le istruzioni nella documentazione AWS, inclusi i seguenti passaggi: • Per la compatibilità con Select cluster, scegli Solo rete, che supporterà Amazon WorkSpace come istanza esterna al cluster ECS. • Scegli di creare un nuovo VPC.	Architetto del cloud

Avvia Amazon WorkSpaces

Operazione	Description	Competenze richieste
Configura Simple AD e avvia Amazon WorkSpaces.	Per effettuare il provisioning di una directory Simple AD per il tuo VPC appena creato e avviare Amazon WorkSpace s, segui le istruzioni nella documentazione AWS.	Architetto del cloud

Configurare AWS Systems Manager per un ambiente ibrido

Operazione	Description	Competenze richieste
Scarica gli script allegati.	Sul computer locale, scaricate i `ssm-activation.json` file `ssm-trust-policy.json` e contenuti nella sezione Allegati.	Architetto del cloud

Aggiungi il ruolo IAM.	Aggiungi variabili di ambiente in base ai tuoi requisiti aziendali.
	`export AWS_DEFAULT_REGION=${AWS_REGION_ID}`
	`export ROLE_NAME=${ECS_TASK_ROLE}`
	`export CLUSTER_NAME=${ECS_CLUSTER_NAME}`
	`export SERVICE_NAME=${ECS_CLUSTER_SERVICE_NAME}`
	Eseguire il seguente comando seguente.
	`aws iam create-role --role-name $ROLE_NAME --assume-role-policy-document file://ssm-trust-policy.json`
		Architetto del cloud

Aggiungi la SSMManaged InstanceCore policy di Amazon al ruolo IAM.	Eseguire il seguente comando seguente. `aws iam attach-role-policy --role-name $ROLE_NAME --policy-`	Architetto del cloud

Operazione	Description	Competenze richieste
	`arn arn:aws:iam::aws:policy/AmazonSSMManagedInstanceCore`	
Aggiungi la policy Amazon EC2 ContainerServicefor EC2 Role al ruolo IAM.	Eseguire il seguente comando seguente. `aws iam attach-role-policy --role-name $ROLE_NAME --policy-arn arn:aws:iam::aws:policy/service-role/AmazonEC2ContainerServiceforEC2Role`	Architetto del cloud
Verifica il ruolo IAM.	Per verificare il ruolo IAM, esegui il comando seguente. `aws iam list-attached-role-policies --role-name $ROLE_NAME`	Architetto del cloud
Attiva Systems Manager.	Eseguire il seguente comando seguente. `aws ssm create-activation --iam-role $ROLE_NAME | tee ssm-activation.json`	Architetto del cloud

Aggiungi WorkSpaces al cluster ECS

Operazione	Description	Competenze richieste
Connect al tuo WorkSpaces.	Per connetterti e configurare i tuoi spazi di lavoro, segui	Sviluppatore di app

Operazione	Description	Competenze richieste
	le istruzioni nella documentazione AWS.	
Scarica lo script di installazione ecs-anywhere.	Nel prompt dei comandi, eseguire il seguente comando . `curl -o "ecs-anywhere-install.sh" "https://amazon-ecs-agent-packages-preview.s3.us-east-1.amazonaws.com/ecs-anywhere-install.sh" && sudo chmod +x ecs-anywhere-install.sh`	Sviluppatore di app
Verifica l'integrità dello script della shell.	(Facoltativo) Eseguite il comando seguente. `curl -o "ecs-anywhere-install.sh.sha256" "https://amazon-ecs-agent-packages-preview.s3.us-east-1.amazonaws.com/ecs-anywhere-install.sh.sha256" && sha256sum -c ecs-anywhere-install.sh.sha256`	Sviluppatore di app

Operazione	Description	Competenze richieste
Aggiungi un repository EPEL su Amazon Linux.	Per aggiungere un repository Extra Packages for Enterprise Linux (EPEL), esegui il comando. `sudo amazon-linux-extras install epel -y`	Sviluppatore di app
Installa Amazon ECS Anywhere.	Per eseguire lo script di installazione, usa il seguente comando. `sudo ./ecs-anywhere-install.sh --cluster $CLUSTER_NAME --activation-id $ACTIVATION_ID --activation-code $ACTIVATION_CODE --region $AWS_REGION`	
Controlla le informazioni sull'istanza dal cluster ECS.	Per controllare le informazioni sulle istanze del cluster Systems Manager ed ECS e convalidare quelle WorkSpaces aggiunte al cluster, esegui il comando seguente dal computer locale. `aws ssm describe-instance-information" && "aws ecs list-container-instances --cluster $CLUSTER_NAME`	Sviluppatore di app

Aggiungi un'attività ECS per WorkSpaces

Operazione	Description	Competenze richieste
Crea un ruolo IAM per l'esecuzione delle attività.	Scarica `task-execution-assume-role.json` e `external-task-definition.json` dalla sezione Allegati. Sul computer locale, esegui il seguente comando. `aws iam --region $AWS_DEFAULT_REGION create-role --role-name $ECS_TASK_EXECUTION_ROLE --assume-role-policy-document file://task-execution-assume-role.json`	Architetto del cloud
Aggiungi la policy al ruolo di esecuzione.	Eseguire il seguente comando seguente. `aws iam --region $AWS_DEFAULT_REGION attach-role-policy --role-name $ECS_TASK_EXECUTION_ROLE --policy-arn arn:aws:iam::aws:policy/service-role/AmazonECSTaskExecutionRolePolicy`	Architetto del cloud
Crea un ruolo da svolgere.	Eseguire il seguente comando seguente.	Architetto del cloud

Operazione	Description	Competenze richieste
	`aws iam --region $AWS_DEFAULT_REGION create-role --role-name $ECS_TASK_EXECUTION_ROLE --assume-role-policy-document file://task-execution-assume-role.json`	
Registra la definizione dell'attività nel cluster.	Sul computer locale, esegui il comando seguente. `aws ecs register-task-definition --cli-input-json file://external-task-definition.json`	Architetto del cloud
Esegui l'attività.	Sul computer locale, esegui il comando seguente. `aws ecs run-task --cluster $CLUSTER_NAME --launch-type EXTERNAL --task-definition nginx`	Architetto del cloud

Operazione	Description	Competenze richieste

Convalida lo stato di esecuzione dell'attività.	Per recuperare l'ID dell'attività, esegui il comando seguente.
	`export TEST_TASKID= $(aws ecs list-tasks -- cluster $CLUSTER_NAME | jq -r '.taskArns[0]')`
	Con l'ID dell'attività, esegui il comando seguente.
	`aws ecs describe-tasks --cluster $CLUSTER_ NAME --tasks ${TEST_TA SKID}`
		Architetto del cloud

Verifica l'attività su WorkSpace .	Per verificare che NGINX sia in esecuzione su WorkSpace , esegui il comando. `curl http://localhost:8 080`	Sviluppatore di app

Risorse correlate

- Cluster ECS
- Configurazione di un ambiente ibrido
- Amazon WorkSpaces
- Simple AD

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Esegui un contenitore Docker dell'API Web ASP.NET Core su un'istanza Amazon Linux EC2

Vijai Anand Ramalingam e Sreelaxmi Pai, Amazon Web Services

Riepilogo

Questo modello è destinato alle persone che stanno iniziando a containerizzare le proprie applicazioni sul cloud Amazon Web Services (AWS). Quando inizi a containerizzare le app sul cloud, di solito non sono configurate piattaforme di orchestrazione dei container. Questo modello ti aiuta a configurare rapidamente l'infrastruttura su AWS per testare le tue applicazioni containerizzate senza bisogno di un'elaborata infrastruttura di orchestrazione dei container.

Il primo passo nel percorso di modernizzazione consiste nel trasformare l'applicazione. Se si tratta di un'applicazione.NET Framework legacy, è necessario innanzitutto modificare il runtime in ASP.NET Core. Quindi, esegui queste operazioni:

- Crea l'immagine del contenitore Docker
- Esegui il contenitore Docker utilizzando l'immagine integrata
- Convalida l'applicazione prima di distribuirla su qualsiasi piattaforma di orchestrazione dei container, come Amazon Elastic Container Service (Amazon ECS) o Amazon Elastic Kubernetes Service (Amazon EKS).

Questo modello copre gli aspetti di creazione, esecuzione e convalida dello sviluppo di applicazioni moderne su un'istanza Amazon Elastic Compute Cloud EC2 (Amazon) Linux.

Prerequisiti e limitazioni

Prerequisiti

- Un account Amazon Web Services (AWS) attivo
- Un ruolo AWS Identity and Access Management (IAM) con accesso sufficiente per creare risorse AWS per questo modello
- Scaricato e installato Visual Studio Community 2022 o versione successiva
- Un progetto.NET Framework modernizzato in ASP.NET Core
- Un repository GitHub

Versioni del prodotto

- Visual Studio Community 2022 o versioni successive

Architecture

Architettura Target

Questo modello utilizza un CloudFormation modello AWS per creare l'architettura ad alta disponibilità mostrata nel diagramma seguente. Un'istanza Amazon EC2 Linux viene avviata in una sottorete privata. AWS Systems Manager Session Manager viene utilizzato per accedere all'istanza privata di Amazon EC2 Linux e per testare l'API in esecuzione nel contenitore Docker.

1. Accesso all'istanza Linux tramite Session Manager

Tools (Strumenti)

Servizi AWS

- Interfaccia a riga di comando AWS: AWS Command Line Interface (AWS CLI) è uno strumento open source per interagire con i servizi AWS tramite comandi nella shell della riga di comando. Con una configurazione minima, puoi eseguire comandi AWS CLI che implementano funzionalità equivalenti a quelle fornite dalla Console di gestione AWS basata su browser.
- Console di gestione AWS: la Console di gestione AWS è un'applicazione Web che comprende e fa riferimento a un'ampia raccolta di console di servizio per la gestione delle risorse AWS. Quando effettui l'accesso per la prima volta, visualizzi la home page della console. La home page fornisce l'accesso a ciascuna console di servizio e offre un unico posto per accedere alle informazioni necessarie per eseguire le attività relative ad AWS.
- AWS Systems Manager Session Manager — Session Manager è una funzionalità di AWS Systems Manager completamente gestita. Con Session Manager, puoi gestire le tue istanze Amazon Elastic Compute Cloud (Amazon EC2). Session Manager fornisce una gestione sicura e verificabile dei nodi senza la necessità di aprire porte in entrata, gestire host bastion o gestire chiavi SSH.

Altri strumenti

- Visual Studio 2022 — Visual Studio 2022 è un ambiente di sviluppo integrato (IDE).

- Docker: Docker è un set di prodotti Platform as a Service (PaaS) che utilizzano la virtualizzazione a livello di sistema operativo per fornire software in contenitori.

Codice

```
FROM mcr.microsoft.com/dotnet/aspnet:5.0 AS base
 WORKDIR /app
EXPOSE 80
EXPOSE 443

FROM mcr.microsoft.com/dotnet/sdk:5.0 AS build
WORKDIR /src
COPY ["DemoNetCoreWebAPI/DemoNetCoreWebAPI.csproj", "DemoNetCoreWebAPI/"]
RUN dotnet restore "DemoNetCoreWebAPI/DemoNetCoreWebAPI.csproj"
COPY . .
WORKDIR "/src/DemoNetCoreWebAPI"
RUN dotnet build "DemoNetCoreWebAPI.csproj" -c Release -o /app/build

FROM build AS publish
RUN dotnet publish "DemoNetCoreWebAPI.csproj" -c Release -o /app/publish

FROM base AS final
WORKDIR /app
COPY --from=publish /app/publish .
ENTRYPOINT ["dotnet", "DemoNetCoreWebAPI.dll"]
```

Epiche

Sviluppa l'API web ASP.NET Core

Operazione	Description	Competenze richieste
Crea un esempio di API Web ASP.NET Core utilizzando Visual Studio.	Per creare un esempio di API web ASP.NET Core, procedi come segue: 1. Apri Visual Studio 2022. 2. Scegliere Create a new project (Crea un nuovo progetto).	Sviluppatore di app

Operazione	Description	Competenze richieste

	3. Seleziona il modello di progetto ASP.NET Core Web API e scegli Avanti.
	4. Per il nome del progetto, inserisci DemoNetCoreWebAPI e scegli Avanti.
	5. Scegli Create (Crea).
	6. Per eseguire il progetto localmente, premi F5.
	7. Verifica che l'endpoint WeatherForecastAPI predefinito restituisca i risultati utilizzando Swagger.
	8. Apri il prompt dei comandi, vai alla cartella del progetto.csproj ed esegui i seguenti comandi per inviare la nuova API web al tuo repository. GitHub
	`git add --all`
	`git commit -m "Initial Version"`
	`git push`	

Operazione	Description	Competenze richieste

Crea un Dockerfile.	Per creare un Dockerfile, esegui una delle seguenti operazioni:
	• Crea il Dockerfile manualmente utilizzando il Dockerfile di esempio nella sezione Codice. In base ai requisiti, seleziona l'immagine di base.NET appropriata. Per informazioni sulle immagini relative a .NET e ASP.NET Core, consulta Docker hub.
	• Crea il Dockerfile utilizzando Visual Studio e Docker Desktop. In Solution Explorer, fai clic con il pulsante destro del mouse sul progetto, scegli Aggiungi -> Docker Support. Per Target OS, seleziona Linux. Assicurati che il nuovo Dockerfile si trovi nello stesso percorso del file della soluzione (.sln).
	
	Per inviare le modifiche al tuo GitHub repository, esegui il comando seguente.
	`git add --all`
	`git commit -m "Dockerfile added"`
		Sviluppatore di app

Operazione	Description	Competenze richieste
	`git push`	

Configurare l'istanza Amazon EC2 Linux

Operazione	Description	Competenze richieste
Configura l'infrastruttura.	Avvia il CloudFormation modello AWS per creare l'infrastruttura, che include quanto segue: • Un cloud privato virtuale (VPC), che utilizza AWS VPC Quick Start, con due sottoreti pubbliche e due private che si estendono su due zone di disponibilità. • Il ruolo IAM richiesto per abilitare AWS Systems Manager. • In una delle sottoreti private, un'istanza demo di Amazon Linux 2 con l'agente SSM più recente. Sebbene questa istanza non abbia alcuna connettività diretta da Internet, è possibile accedervi in modo sicuro utilizzando AWS Systems Manager Session Manager senza richiedere un host bastion.	Sviluppatore di app, amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste
	Note Amazon Linux 2 sta per terminare il supporto. Per ulteriori informazioni, consulta Amazon Linux 2 FAQs. Per ulteriori informazioni sull'accesso a un' EC2 istanza Amazon privata utilizzando Session Manager senza richiedere un host bastion, consulta il post sul blog Toward a bastion-less world.	

Operazione	Description	Competenze richieste
Accedi all'istanza Amazon EC2 Linux.	Per connetterti all'istanza Amazon EC2 Linux nella sottorete privata, procedi come segue: 1. Apri la EC2 console Amazon. 2. Nel riquadro di navigazione, scegliere Instances (Istanze). 3. Seleziona l'istanza demo di Amazon Linux 2 e scegli Connect. **Note** Amazon Linux 2 sta per terminare il supporto. Per ulteriori informazioni, consulta Amazon Linux 2 FAQs. 4. Scegli Session Manager. 5. Scegli Connect per aprire una nuova finestra di terminale. 6. Eseguire il seguente comando seguente. `sudo su`	Sviluppatore di app

Operazione	Description	Competenze richieste

Installa e avvia Docker.	Per installare e avviare Docker nell'istanza Amazon EC2 Linux, procedi come segue:
	1. Per installare Docker, esegui il comando seguente.
	`yum install -y docker`
	2. Per avviare il servizio Docker, esegui il comando seguente.
	`service docker start`
	3. Per verificare l'installazione di Docker, esegui il comando seguente.
	`docker info`
		Sviluppatore di app, amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste

Installa Git e clona il repository.	Per installare Git sull'istanza Amazon EC2 Linux e clonare il repository GitHub, procedi come segue.
	1. Per installare Git, esegui il seguente comando.
	`yum install git -y`
	2. Per clonare il repository, esegui il comando seguente.
	`git clone https://github.com/<username>/<repo-name>.git`
	3. Per accedere al Dockerfile, esegui il comando seguente.
	`cd <repo-name>/DemoNetCoreWebAPI/`
		Sviluppatore di app, amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste

Crea ed esegui il contenitore Docker.	Per creare l'immagine Docker ed eseguire il contenitore all'interno dell'istanza Amazon EC2 Linux, procedi come segue:
	1. Per creare l'immagine Docker, esegui il comando seguente.
	`docker build -t aspnetcorewebapiimage -f Dockerfile .`
	2. Per visualizzare tutte le immagini Docker, esegui il comando seguente.
	`docker images`
	3. Per creare ed eseguire il contenitore, esegui il comando seguente.
	`docker run -d -p 80:80 --name aspnetcorewebapicontainer aspnetcorewebapiimage`
		Sviluppatore di app, amministratore AWS, AWS DevOps

Testa l'API web

Operazione	Description	Competenze richieste
Testa l'API web usando il comando curl.	Per testare l'API web, esegui il comando seguente.	Sviluppatore di app

Operazione	Description	Competenze richieste
	`curl -X GET "http://localhost/WeatherForecast" -H "accept: text/plain"` Verifica la risposta dell'API. **Note** Puoi ottenere i comandi curl per ogni endpoint da Swagger quando lo esegui localmente.	

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Eliminare tutte le risorse.	Elimina lo stack per rimuovere tutte le risorse. In questo modo non ti verranno addebitati costi per i servizi che non utilizzi.	Amministratore AWS, AWS DevOps

Risorse correlate

- Connect alla propria istanza Linux da Windows utilizzando PuTTY
- Crea un'API web con ASP.NET Core
- Verso un mondo senza bastioni

Esegui carichi di lavoro con stato con storage persistente dei dati utilizzando Amazon EFS su Amazon EKS con AWS Fargate

Ricardo Morais, Rodrigo Bersa e Lucio Pereira, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni per abilitare Amazon Elastic File System (Amazon EFS) come dispositivo di storage per contenitori in esecuzione su Amazon Elastic Kubernetes Service (Amazon EKS) utilizzando AWS Fargate per il provisioning delle risorse di calcolo.

La configurazione descritta in questo modello segue le migliori pratiche di sicurezza e fornisce sicurezza a riposo e sicurezza in transito per impostazione predefinita. Per crittografare il tuo file system Amazon EFS, utilizza una chiave AWS Key Management Service (AWS KMS), ma puoi anche specificare un alias chiave che esegua il processo di creazione di una chiave KMS.

Puoi seguire i passaggi di questo schema per creare uno spazio dei nomi e un profilo Fargate per un'applicazione proof-of-concept (PoC), installare il driver Amazon EFS Container Storage Interface (CSI) utilizzato per integrare il cluster Kubernetes con Amazon EFS, configurare la classe di storage e distribuire l'applicazione PoC. Questi passaggi portano a un file system Amazon EFS condiviso tra più carichi di lavoro Kubernetes, in esecuzione su Fargate. Lo schema è accompagnato da script che automatizzano questi passaggi.

È possibile utilizzare questo modello se si desidera la persistenza dei dati nelle applicazioni containerizzate ed evitare la perdita di dati durante le operazioni di scalabilità. Esempio:

- DevOps strumenti — Uno scenario comune è lo sviluppo di uno strumento di integrazione e distribuzione continua (strumento). CI/CD) strategy. In this case, you can use Amazon EFS as a shared file system to store configurations among different instances of the CI/CD tool or to store a cache (for example, an Apache Maven repository) for pipeline stages among different instances of the CI/CD
- Server Web: uno scenario comune consiste nell'utilizzare Apache come server Web HTTP. Puoi usare Amazon EFS come file system condiviso per archiviare file statici condivisi tra diverse istanze del server Web. In questo scenario di esempio, le modifiche vengono applicate direttamente al file system anziché inserire file statici in un'immagine Docker.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un cluster Amazon EKS esistente con Kubernetes versione 1.17 o successiva (testato fino alla versione 1.27)
- Un file system Amazon EFS esistente per associare un Kubernetes StorageClass e fornire i file system in modo dinamico
- Autorizzazioni di amministrazione del cluster
- Contesto configurato per puntare al cluster Amazon EKS desiderato

Limitazioni

- Ci sono alcune limitazioni da considerare quando usi Amazon EKS con Fargate. Ad esempio, l'uso di alcuni costrutti Kubernetes, come DaemonSets i contenitori privilegiati, non è supportato. Per ulteriori informazioni sulle limitazioni di Fargate, consulta le considerazioni su AWS Fargate nella documentazione di Amazon EKS.
- Il codice fornito con questo pattern supporta le workstation che eseguono Linux o macOS.

Versioni del prodotto

- AWS Command Line Interface (AWS CLI) versione 2 o successiva
- Driver Amazon EFS CSI versione 1.0 o successiva (testato fino alla versione 2.4.8)
- eksctl versione 0.24.0 o successiva (testato fino alla versione 0.158.0)
- jq versione 1.6 o successiva
- kubectl versione 1.17 o successiva (testata fino alla versione 1.27)
- Kubernetes versione 1.17 o successiva (testato fino alla versione 1.27)

Architecture

L'architettura di destinazione è composta dalla seguente infrastruttura:

- Un cloud privato virtuale (VPC)
- Due zone di disponibilità
- Una sottorete pubblica con un gateway NAT che fornisce l'accesso a Internet

- Una sottorete privata con un cluster Amazon EKS e target di montaggio Amazon EFS (noti anche come punti di montaggio)
- Amazon EFS a livello di VPC

Di seguito è riportata l'infrastruttura ambientale per il cluster Amazon EKS:

- Profili AWS Fargate che supportano i costrutti Kubernetes a livello di namespace
- Uno spazio dei nomi Kubernetes con:
  - Due pod applicativi distribuiti tra le zone di disponibilità
  - Una dichiarazione di volume persistente (PVC) associata a un volume persistente (PV) a livello di cluster
- Un PV a livello di cluster associato al PVC nello spazio dei nomi e che punta alle destinazioni di montaggio di Amazon EFS nella sottorete privata, all'esterno del cluster

Tools (Strumenti)

Servizi AWS

- AWS Command Line Interface (AWS CLI) è uno strumento open source che puoi usare per interagire con i servizi AWS dalla riga di comando.
- Amazon Elastic File System (Amazon EFS) ti aiuta a creare e configurare file system condivisi nel cloud AWS. In questo modello, fornisce un file system semplice, scalabile, completamente gestito e condiviso da utilizzare con Amazon EKS.
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire Kubernetes su AWS senza dover installare o gestire i tuoi cluster.
- AWS Fargate è un motore di elaborazione serverless per Amazon EKS. Crea e gestisce risorse di calcolo per le tue applicazioni Kubernetes.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.

Altri strumenti

- Docker è un insieme di prodotti Platform as a Service (PaaS) che utilizzano la virtualizzazione a livello di sistema operativo per fornire software in container.

- eksctl è un'utilità da riga di comando per la creazione e la gestione di cluster Kubernetes su Amazon EKS.
- kubectl è un'interfaccia a riga di comando che ti aiuta a eseguire comandi sui cluster Kubernetes.
- jq è uno strumento a riga di comando per l'analisi di JSON.

Codice

Il codice per questo pattern è fornito nella configurazione di GitHub persistenza con Amazon EFS su Amazon EKS utilizzando il repository AWS Fargate. Gli script sono organizzati da epic, nelle cartelle `epic01` Through`epic06`, corrispondenti all'ordine nella sezione Epics di questo schema.

Best practice

L'architettura di destinazione include i seguenti servizi e componenti e segue le best practice di AWS Well-Architected Framework:

- Amazon EFS, che fornisce un file system NFS elastico semplice, scalabile e completamente gestito. Viene utilizzato come file system condiviso tra tutte le repliche dell'applicazione PoC in esecuzione nei pod, distribuiti nelle sottoreti private del cluster Amazon EKS scelto.
- Una destinazione di montaggio Amazon EFS per ogni sottorete privata. Ciò fornisce ridondanza per zona di disponibilità all'interno del cloud privato virtuale (VPC) del cluster.
- Amazon EKS, che esegue i carichi di lavoro Kubernetes. È necessario effettuare il provisioning di un cluster Amazon EKS prima di utilizzare questo modello, come descritto nella sezione Prerequisiti.
- AWS KMS, che fornisce la crittografia a riposo per i contenuti archiviati nel file system Amazon EFS.
- Fargate, che gestisce le risorse di elaborazione per i container in modo che tu possa concentrarti sui requisiti aziendali anziché sul carico dell'infrastruttura. Il profilo Fargate viene creato per tutte le sottoreti private. Fornisce ridondanza per zona di disponibilità all'interno del cloud privato virtuale (VPC) del cluster.
- Kubernetes Pods, per verificare che i contenuti possano essere condivisi, consumati e scritti da diverse istanze di un'applicazione.

Epiche

Esegui il provisioning di un cluster Amazon EKS (opzionale)

Operazione	Description	Competenze richieste
Crea un cluster Amazon EKS.	**Note** Se hai già un cluster distribuito, passa alla prossima epopea. Crea un cluster Amazon EKS nel tuo account AWS esistente. Nella GitHub directory, utilizza uno dei modelli per distribuire un cluster Amazon EKS utilizzando Terraform o eksctl. Per ulteriori informazioni, consulta Creazione di un cluster Amazon EKS nella documentazione di Amazon EKS. Nel modello Terraform, ci sono anche esempi che mostrano come: collegare i profili Fargate al cluster Amazon EKS, creare un file system Amazon EFS e distribuire il driver CSI Amazon	Amministratore AWS, amministratore Terraform o eksctl, amministratore Kubernetes

Operazione	Description	Competenze richieste
	EFS nel cluster Amazon EKS.	

Operazione	Description	Competenze richieste

Esporta variabili di ambiente.	Esegui lo script env.sh. Ciò fornisce le informazioni richieste nei passaggi successivi.
	`source ./scripts/env.sh`
	`Inform the AWS Account ID: <13-digit-account-id>`
	`Inform your AWS Region: <aws-Region-code>`
	`Inform your Amazon EKS Cluster Name: <amazon-eks-cluster-name>`
	`Inform the Amazon EFS Creation Token: <self-genereated-uuid>`
	Se non ancora indicato, puoi ottenere tutte le informazioni richieste sopra con i seguenti comandi CLI.
	`# ACCOUNT ID`
	`aws sts get-caller-identity --query "Account" --output text`
	`# REGION CODE`
	`aws configure get region`
	`# CLUSTER EKS NAME`
	`aws eks list-clusters --query "clusters" --output text`
		Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	`# GENERATE EFS TOKEN` `uuidgen`	

Crea un namespace Kubernetes e un profilo Fargate collegato

Operazione	Description	Competenze richieste

Crea uno spazio dei nomi Kubernetes e un profilo Fargate per i carichi di lavoro delle applicazioni.	Crea uno spazio dei nomi per ricevere i carichi di lavoro delle applicazioni che interagiscono con Amazon EFS. Eseguire lo script `create-k8s-ns-and-linked-fargate-profile.sh` . Puoi scegliere di utilizzare un nome di namespace personalizzato o lo spazio dei nomi fornito di default. `poc-efs-eks-fargate`
	
	Con un nome di namespace dell'applicazione personalizzato:
	
	`export $APP_NAMESPACE=<CUSTOM_NAME>`
	`./scripts/epic01/create-k8s-ns-and-linked-fargate-profile.sh \`
	`-c "$CLUSTER_NAME" -n "$APP_NAMESPACE"`
		Utente Kubernetes con autorizzazioni concesse

Operazione	Description	Competenze richieste

	Senza un nome di namespace dell'applicazione personalizzato:
	`./scripts/epic01/create-k8s-ns-and-linked-fargate-profile.sh \`
	`-c "$CLUSTER_NAME"`
	`$CLUSTER_NAME` dov'è il nome del tuo cluster Amazon EKS. Il `-n <NAMESPACE>` parametro è facoltativo; se non viene informato, verrà fornito un nome di namespace generato di default.	

Creare un file system Amazon EFS

Operazione	Description	Competenze richieste
Genera un token unico.	Amazon EFS richiede un token di creazione per garantire un funzionamento idempotente (la chiamata all'operazione con lo stesso token di creazione non ha alcun effetto). Per soddisfare questo requisito, è necessario generare un token univoco utilizzando una tecnica disponibile. Ad esempio, è possibile generare un identificatore univoco universale	Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	(UUID) da utilizzare come token di creazione.	

Operazione	Description	Competenze richieste
Crea un file system Amazon EFS.	Crea il file system per ricevere i file di dati letti e scritti dai carichi di lavoro dell'appl icazione. È possibile creare un file system crittografato o non crittografato. (Come procedura ottimale, il codice di questo modello crea un sistema crittografato per abilitare la crittografia a riposo per impostazione predefinita.) Puoi utilizzare una chiave AWS KMS unica e simmetrica per crittogra fare il tuo file system. Se non viene specificata una chiave personalizzata, viene utilizzata una chiave gestita AWS. Utilizza lo script create-ef s.sh per creare un file system Amazon EFS crittografato o non crittografato, dopo aver generato un token univoco per Amazon EFS. Con crittografia inattiva, senza chiave KMS: `./scripts/epic02/c reate-efs.sh \ -c "$CLUSTER_NAME" \ -t "$EFS_CRE ATION_TOKEN"`	Amministratore di sistema AWS

Operazione	Description	Competenze richieste

	dove `$CLUSTER_NAME` è il nome del tuo cluster Amazon EKS ed `$EFS_CREATION_TOKEN` è un token di creazione univoco per il file system.
	
	Con crittografia inattiva, con una chiave KMS:
	
	`./scripts/epic02/create-efs.sh \ -c "$CLUSTER_NAME" \ -t "$EFS_CREATION_TOKEN" \ -k "$KMS_KEY_ALIAS"`
	
	dove `$CLUSTER_NAME` è il nome del cluster Amazon EKS, `$EFS_CREATION_TOKEN` è un token di creazione univoco per il file system ed `$KMS_KEY_ALIAS` è l'alias per la chiave KMS.
	
	Senza crittografia:
	
	`./scripts/epic02/create-efs.sh -d \ -c "$CLUSTER_NAME" \ -t "$EFS_CREATION_TOKEN"`	

Operazione	Description	Competenze richieste
	dove `$CLUSTER_NAME` è il nome del tuo cluster Amazon EKS, `$EFS_CREATION_TOKEN` è un token di creazione univoco per il file system e `-d` disabilita la crittografia a riposo.	
Creare un gruppo di sicurezza.	Crea un gruppo di sicurezza per consentire al cluster Amazon EKS di accedere al file system Amazon EFS.	Amministratore di sistema AWS
Aggiorna la regola in entrata per il gruppo di sicurezza.	Aggiorna le regole in entrata del gruppo di sicurezza per consentire il traffico in entrata per le seguenti impostazioni: • Protocollo TCP: porta 2049 • Fonte: intervalli di blocchi CIDR per le sottoreti private nel VPC che contiene il cluster Kubernetes	Amministratore di sistema AWS
Aggiungi una destinazione di montaggio per ogni sottorete privata.	Per ogni sottorete privata del cluster Kubernetes, crea una destinazione di montaggio per il file system e il gruppo di sicurezza.	Amministratore di sistema AWS

Installa i componenti Amazon EFS nel cluster Kubernetes

Operazione	Description	Competenze richieste
Implementa il driver CSI di Amazon EFS.	Implementa il driver CSI Amazon EFS nel cluster. Il driver effettua il provision ing dello storage in base alle dichiarazioni di volume persistenti create dalle applicazioni. Esegui lo `create-k8s-efs-csi-sc.sh` script per distribuire il driver CSI di Amazon EFS e la classe di storage nel cluster. `./scripts/epic03/create-k8s-efs-csi-sc.sh` Questo script utilizza l'`kubectl`utilità, quindi assicurati che il contesto sia stato configurato e punti al cluster Amazon EKS desiderato.	Utente Kubernetes con autorizzazioni concesse
Implementare la classe di archiviazione.	Implementa la classe di storage nel cluster per il provisioner Amazon EFS (efs.csi.aws.com).	Utente Kubernetes con autorizzazioni concesse

Installa l'applicazione PoC nel cluster Kubernetes

Operazione	Description	Competenze richieste
Distribuisci il volume persistente.	Implementa il volume persistente e collegalo alla classe di storage creata e all'ID del file system Amazon EFS. L'applicazione utilizza il volume persistente per leggere e scrivere contenuti . È possibile specificare qualsiasi dimensione per il volume persistente nel campo di archiviazione. Kubernetes richiede questo campo, ma poiché Amazon EFS è un file system elastico, non impone alcuna capacità del file system. Puoi distribuire il volume persistente con o senza crittografia. (Il driver CSI di Amazon EFS abilita la crittografia per impostazione predefinita, come best practice.) Esegui lo `deploy-poc-app.sh` script per distribuire il volume persistente, l'attestazione del volume persistente e i due carichi di lavoro. Con crittografia in transito: `./scripts/epic04/deploy-poc-app.sh \`	Utente Kubernetes con autorizzazioni concesse

Operazione	Description	Competenze richieste

	`-t "$EFS_CRE ATION_TOKEN"`
	
	`$EFS_CREATION_TOKE N` dov'è il token di creazione univoco per il file system.
	
	Senza crittografia in transito:
	
	`./scripts/epic04/d eploy-poc-app.sh -d \ -t "$EFS_CRE ATION_TOKEN"`
	
	dove `$EFS_CREA TION_TOKEN` è il token di creazione univoco per il file system e `-d` disabilita la crittografia in transito.	

Implementa la dichiarazione di volume persistente richiesta dall'applicazione.	Implementate la dichiarazione di volume persistente richiesta dall'applicazione e collegate la alla classe di archiviazione. Utilizza la stessa modalità di accesso del volume persisten te creato in precedenza. È possibile specificare qualsiasi dimensione per l'attesta zione del volume persistente nel campo di archiviazione. Kubernetes richiede questo campo, ma poiché Amazon EFS è un file system elastico, non impone alcuna capacità del file system.	Utente Kubernetes con autorizzazioni concesse

Operazione	Description	Competenze richieste
Distribuisci il carico di lavoro 1.	Distribuisci il pod che rappresenta il carico di lavoro 1 dell'applicazione. Questo carico di lavoro scrive il contenuto nel file. `/data/out 1.txt`	Utente Kubernetes con autorizzazioni concesse
Implementa il carico di lavoro 2.	Implementa il pod che rappresenta il carico di lavoro 2 dell'applicazione. Questo carico di lavoro scrive il contenuto nel file. `/data/out 2.txt`	Utente Kubernetes con autorizzazioni concesse

Convalida la persistenza, la durabilità e la condivisibilità del file system

Operazione	Description	Competenze richieste
Controlla lo stato di`PersistentVolume` .	Immettere il seguente comando per verificare lo stato di`PersistentVolume` . `kubectl get pv` Per un esempio di output, vedere la sezione Informazioni aggiuntive.	Utente Kubernetes con autorizzazioni concesse
Controlla lo stato di. `PersistentVolumeCl aim`	Immettere il seguente comando per verificare lo stato di`PersistentVolumeCl aim` .	Utente Kubernetes con autorizzazioni concesse

Operazione	Description	Competenze richieste
	`kubectl -n poc-efs-eks-fargate get pvc` Per un esempio di output, vedere la sezione Informazioni aggiuntive.	

Verifica che il workload 1 possa scrivere sul file system.	Immetti il comando seguente per convalidare il carico di lavoro 1 su cui sta scrivendo. `/data/out1.txt`
	
	`kubectl exec -ti poc-app1 -n poc-efs-eks-fargate -- tail -f /data/out1.txt`
	
	I risultati sono simili ai seguenti:
	
	`...`
	`Thu Sep  3 15:25:07 UTC 2023 - PoC APP 1`
	`Thu Sep  3 15:25:12 UTC 2023 - PoC APP 1`
	`Thu Sep  3 15:25:17 UTC 2023 - PoC APP 1`
	`...`
		Utente Kubernetes con autorizzazioni concesse

Operazione	Description	Competenze richieste

Verifica che Workload 2 sia in grado di scrivere sul file system.	Immetti il comando seguente per convalidare il workload 2 su cui sta scrivendo. `/data/out2.txt`
	`kubectl -n $APP_NAME SPACE exec -ti poc-app2 -- tail -f /data/out 2.txt`
	I risultati sono simili ai seguenti:
	`... Thu Sep 3 15:26:48 UTC 2023 - PoC APP 2 Thu Sep 3 15:26:53 UTC 2023 - PoC APP 2 Thu Sep 3 15:26:58 UTC 2023 - PoC APP 2 ...`
		Utente Kubernetes con autorizzazioni concesse

Operazione	Description	Competenze richieste

Verifica che il carico di lavoro 1 sia in grado di leggere il file scritto da workload 2.	Immetti il seguente comando per verificare che il carico di lavoro 1 sia in grado di leggere il `/data/out2.txt` file scritto dal carico di lavoro 2.
	`kubectl exec -ti poc-app1 -n poc-efs-eks-fargate -- tail -n 3 /data/out2.txt`
	I risultati sono simili ai seguenti:
	`... Thu Sep 3 15:26:48 UTC 2023 - PoC APP 2 Thu Sep 3 15:26:53 UTC 2023 - PoC APP 2 Thu Sep 3 15:26:58 UTC 2023 - PoC APP 2 ...`
		Utente Kubernetes con autorizzazioni concesse

Operazione	Description	Competenze richieste

Verifica che il carico di lavoro 2 sia in grado di leggere il file scritto dal carico di lavoro 1.	Immetti il seguente comando per verificare che il carico di lavoro 2 sia in grado di leggere il `/data/out1.txt` file scritto dal carico di lavoro 1.
	`kubectl -n $APP_NAMESPACE exec -ti poc-app2 -- tail -n 3 /data/out1.txt`
	I risultati sono simili ai seguenti:
	`... Thu Sep 3 15:29:22 UTC 2023 - PoC APP 1 Thu Sep 3 15:29:27 UTC 2023 - PoC APP 1 Thu Sep 3 15:29:32 UTC 2023 - PoC APP 1 ...`
		Utente Kubernetes con autorizzazioni concesse

Operazione	Description	Competenze richieste

Verifica che i file vengano conservati dopo aver rimosso i componenti dell'applicazione.	Successivamente, utilizzat e uno script per rimuovere i componenti dell'appl icazione (persistent volume, persistent volume claim e pods) e verificare che i file `/data/out2.txt` vengano conservati nel file `/data/out1.txt` system. Eseguire lo script `validate-efs-content.sh` utilizzando il comando seguente.
	
	`./scripts/epic05/validate-efs-content.sh \`
	`  -t "$EFS_CREATION_TOKEN"`
	
	`$EFS_CREATION_TOKEN` dov'è il token di creazione univoco per il file system.
	
	I risultati sono simili ai seguenti:
	
	`pod/poc-app-validation created`
	`Waiting for pod get Running state...`
	`Waiting for pod get Running state...`
	`Waiting for pod get Running state...`
	`Results from execution of 'find /data' on`
		Utente Kubernetes con autorizzazioni concesse, amministratore di sistema

Operazione	Description	Competenze richieste
	validation process pod: /data /data/out2.txt /data/out1.txt	

Monitora le operazioni

Operazione	Description	Competenze richieste
Monitora i registri delle applicazioni.	Nell'ambito di un'operazione che dura il secondo giorno, spedisci i log delle applicazioni ad Amazon CloudWatch per il monitoraggio.	Amministratore di sistema AWS, utente Kubernetes con autorizzazioni concesse
Monitora i contenitori Amazon EKS e Kubernetes con Container Insights.	Nell'ambito di un'operaz ione del secondo giorno, monitora i sistemi Amazon EKS e Kubernetes utilizzando Amazon Container Insights. CloudWatch Questo strumento raccoglie, aggrega e riepiloga i parametri delle applicazioni containerizzate a diversi livelli e dimensioni. Per ulteriori informazioni, consulta la sezione Risorse correlate.	Amministratore di sistema AWS, utente Kubernetes con autorizzazioni concesse
Monitora Amazon EFS con CloudWatch.	Nell'ambito di un'operazione del secondo giorno, monitora i file system utilizzando Amazon CloudWatch, che raccoglie ed elabora i dati grezzi da Amazon EFS in metriche	Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	leggibili quasi in tempo reale. Per ulteriori informazioni, consulta la sezione Risorse correlate.	

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Pulisci tutte le risorse create per il pattern.	Dopo aver completato questo schema, pulisci tutte le risorse per evitare di incorrere in costi AWS. Esegui lo `clean-up-resources.sh` script per rimuovere tutte le risorse dopo aver finito di utilizzare l'applicazione PoC. Completate una delle seguenti opzioni. Con crittografia inattiva, con una chiave KMS: `./scripts/epic06/clean-up-resources.sh \ -c "$CLUSTER_NAME" \ -t "$EFS_CREATION_TOKEN" \ -k "$KMS_KEY_ALIAS"` dove `$CLUSTER_NAME` è il nome del cluster Amazon EKS, `$EFS_CREATION_TOKEN` è il token di creazione per il file system ed	Utente Kubernetes con autorizzazioni concesse, amministratore di sistema

Operazione	Description	Competenze richieste
	`$KMS_KEY_ALIAS` è l'alias per la chiave KMS. Senza crittografia a riposo: `./scripts/epic06/clean-up-resources.sh \ -c "$CLUSTER_NAME" \ -t "$EFS_CREATION_TOKEN"` dove `$CLUSTER_NAME` è il nome del cluster Amazon EKS ed `$EFS_CREATION_TOKEN` è il token di creazione per il file system.	

Risorse correlate

Riferimenti

- AWS Fargate per Amazon EKS ora supporta Amazon EFS (annuncio)
- Come acquisire i log delle applicazioni quando si utilizza Amazon EKS su AWS Fargate (post sul blog)
- Utilizzo di Container Insights (CloudWatch documentazione Amazon)
- Configurazione di Container Insights su Amazon EKS e Kubernetes (documentazione Amazon) CloudWatch
- Metriche di Amazon EKS e Kubernetes Container Insights (documentazione Amazon) CloudWatch
- Monitoraggio di Amazon EFS con Amazon CloudWatch (documentazione Amazon EFS)

GitHub tutorial ed esempi

- Provisioning statico

- Crittografia in transito
- Accesso al file system da più pod
- Consumo di Amazon EFS in StatefulSets
- Montaggio dei sottopercorsi
- Utilizzo dei punti di accesso Amazon EFS
- Progetti Amazon EKS per Terraform

Strumenti necessari

- Installazione della versione 2 dell'interfaccia a riga di comando di AWS
- Installazione di eksctl
- Installare kubectl
- Installazione di jq

Informazioni aggiuntive

Di seguito è riportato un esempio di output del `kubectl get pv` comando.

```
NAME         CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM
            STORAGECLASS   REASON   AGE
poc-app-pv   1Mi        RWX            Retain           Bound    poc-efs-eks-fargate/
poc-app-pvc   efs-sc                  3m56s
```

Di seguito è riportato un esempio di output del `kubectl -n poc-efs-eks-fargate get pvc` comando.

```
NAME          STATUS   VOLUME       CAPACITY   ACCESS MODES   STORAGECLASS   AGE
poc-app-pvc   Bound    poc-app-pv   1Mi        RWX            efs-sc         4m34s
```

Configura la scalabilità automatica basata sugli eventi in Amazon EKS utilizzando Amazon EKS Pod Identity e KEDA

Dipen Desai, Abhay Diwan, Kamal Joshi e Mahendra Revanasiddappa, Amazon Web Services

Riepilogo

Le piattaforme di orchestrazione, come Amazon Elastic Kubernetes Service (Amazon EKS), hanno semplificato la gestione del ciclo di vita delle applicazioni basate su container. Questo aiuta le organizzazioni a concentrarsi sulla creazione, la protezione, il funzionamento e la manutenzione di applicazioni basate su contenitori. Man mano che le implementazioni basate sugli eventi diventano più comuni, le organizzazioni scalano sempre più spesso le implementazioni di Kubernetes in base a varie fonti di eventi. Questo metodo, combinato con la scalabilità automatica, può portare a significativi risparmi sui costi fornendo risorse di elaborazione su richiesta e una scalabilità efficiente adattata alla logica dell'applicazione.

KEDA è un autoscaler basato su eventi basato su Kubernetes. KEDA ti aiuta a scalare qualsiasi contenitore in Kubernetes in base al numero di eventi che devono essere elaborati. È leggero e si integra con qualsiasi cluster Kubernetes. Funziona anche con componenti Kubernetes standard, come Horizontal Pod Autoscaling (HPA). Offre anche KEDA TriggerAuthentication, una funzionalità che consente di delegare l'autenticazione. Consente di descrivere i parametri di autenticazione separati dai contenitori ScaledObject e dai contenitori di distribuzione.

AWS fornisce ruoli AWS Identity and Access Management (IAM) che supportano diverse opzioni di implementazione di Kubernetes, tra cui Amazon EKS, Amazon EKS Anywhere Servizio Red Hat OpenShift su AWS (ROSA) e cluster Kubernetes autogestiti su Amazon Elastic Compute Cloud (Amazon). EC2 Questi ruoli utilizzano costrutti IAM, come i provider di identità OpenID Connect (OIDC) e le policy di fiducia IAM, per operare in ambienti diversi senza affidarsi direttamente ai servizi Amazon EKS o. APIs Per ulteriori informazioni, consulta i ruoli IAM per gli account di servizio nella documentazione di Amazon EKS.

Amazon EKS Pod Identity semplifica il processo con cui gli account di servizio Kubernetes assumono ruoli IAM senza richiedere provider OIDC. Offre la possibilità di gestire le credenziali per le tue applicazioni. Invece di creare e distribuire AWS le tue credenziali ai contenitori o utilizzare il ruolo dell' EC2 istanza Amazon, associ un ruolo IAM a un account di servizio Kubernetes e configuri i tuoi Pods per utilizzare l'account di servizio. Questo ti aiuta a utilizzare un ruolo IAM su più cluster e semplifica la gestione delle policy abilitando il riutilizzo delle politiche di autorizzazione tra i ruoli IAM.

Implementando KEDA con Amazon EKS Pod Identity, le aziende possono ottenere una scalabilità automatica efficiente basata sugli eventi e una gestione semplificata delle credenziali. Le applicazioni si scalano in base alla domanda, il che ottimizza l'utilizzo delle risorse e riduce i costi.

Questo modello ti aiuta a integrare Amazon EKS Pod Identity con KEDA. Mostra come utilizzare l'account del `keda-operator` servizio e delegare l'autenticazione con. `TriggerAuthentication` Descrive inoltre come impostare una relazione di trust tra un ruolo IAM per l'operatore KEDA e un ruolo IAM per l'applicazione. Questa relazione di fiducia consente a KEDA di monitorare i messaggi nelle code degli eventi e di regolare il ridimensionamento per gli oggetti Kubernetes di destinazione.

Prerequisiti e limitazioni

Prerequisiti

- AWS Command Line Interface (AWS CLI) versione 2.13.17 o successiva, installata
- Python versione 3.11.5 o successiva, installata
- AWS SDK per Python (Boto3) versione 1.34.135 o successiva, installata
- Helm versione 3.12.3 o successiva, installata
- kubectl versione 1.25.1 o successiva, installata
- Docker Engine versione 26.1.1 o successiva, installata
- Creazione di un cluster Amazon EKS versione 1.24 o successiva
- Prerequisiti per la creazione dell'agente Amazon EKS Pod Identity, soddisfatti

Limitazioni

- È necessario stabilire una relazione di fiducia tra il `keda-operator` ruolo e il `keda-identity` ruolo. Le istruzioni sono fornite nella sezione Epics di questo modello.

Architecture

In questo modello, crei le seguenti AWS risorse:

- Repository Amazon Elastic Container Registry (Amazon ECR): in questo modello, questo repository viene denominato. `keda-pod-identity-registry` Questo repository privato viene utilizzato per archiviare le immagini Docker dell'applicazione di esempio.
- Coda Amazon Simple Queue Service (Amazon SQS): in questo modello, questa coda viene denominata. `event-messages-queue` La coda funge da buffer dei messaggi che raccoglie e

archivia i messaggi in arrivo. KEDA monitora le metriche della coda, come il numero di messaggi o la lunghezza della coda, e ridimensiona automaticamente l'applicazione in base a queste metriche.

- Ruolo IAM per l'applicazione: in questo modello, questo ruolo viene denominato. `keda-identity` Il `keda-operator` ruolo assume questo ruolo. Questo ruolo consente l'accesso alla coda di Amazon SQS.
- Ruolo IAM per l'operatore KEDA: in questo modello, questo ruolo è denominato. `keda-operator` L'operatore KEDA utilizza questo ruolo per effettuare le chiamate AWS API richieste. Questo ruolo dispone delle autorizzazioni necessarie per assumerlo`keda-identity`. A causa della relazione di fiducia tra i ruoli `keda-operator` e i `keda-identity` ruoli, il `keda-operator` ruolo dispone delle autorizzazioni Amazon SQS.

Tramite le risorse personalizzate `TriggerAuthentication` e `ScaledObject` Kubernetes, l'operatore utilizza il `keda-identity` ruolo per connettersi a una coda Amazon SQS. In base alla dimensione della coda, KEDA ridimensiona automaticamente la distribuzione dell'applicazione. Aggiunge 1 pod ogni 5 messaggi non letti nella coda. Nella configurazione predefinita, se non ci sono messaggi non letti nella coda di Amazon SQS, l'applicazione viene ridimensionata fino a 0 pod. L'operatore KEDA monitora la coda a intervalli specificati dall'utente.

L'immagine seguente mostra come utilizzi Amazon EKS Pod Identity per fornire al `keda-operator` ruolo un accesso sicuro alla coda Amazon SQS.

Il diagramma mostra il flusso di lavoro seguente:

1. Installa l'agente Amazon EKS Pod Identity nel cluster Amazon EKS.
2. Implementa l'operatore KEDA nello spazio dei nomi KEDA nel cluster Amazon EKS.
3. Crei i ruoli `keda-operator` e `keda-identity` IAM nella destinazione. Account AWS
4. Stabilisci una relazione di fiducia tra i ruoli IAM.
5. L'applicazione viene distribuita nel `security` namespace.
6. L'operatore KEDA esegue il polling dei messaggi in una coda Amazon SQS.
7. KEDA avvia l'HPA, che ridimensiona automaticamente l'applicazione in base alla dimensione della coda.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire AWS Kubernetes senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon Simple Queue Service (Amazon SQS) fornisce una coda ospitata sicura, durevole e disponibile che ti aiuta a integrare e disaccoppiare sistemi e componenti software distribuiti.

Altri strumenti

- KEDA è un autoscaler basato su eventi basato su Kubernetes.

Deposito di codice

Il codice per questo pattern è disponibile nella scalabilità GitHub automatica basata sugli eventi utilizzando EKS Pod Identity e il repository KEDA.

Best practice

Consigliamo di seguire queste best practices:

- Best practice di Amazon EKS
- Best practice per la sicurezza in IAM
- Best practice di Amazon SQS

Epiche

Crea risorse AWS

Operazione	Description	Competenze richieste
Crea il ruolo IAM per l'operatore KEDA.	1. Accedi a Console di gestione AWS, quindi apri la console IAM. 2. Nel pannello di navigazione, seleziona Roles (Ruoli). 3. Selezionare Create role (Crea ruolo). 4. Scegli il tipo di ruolo Custom trust policy (Policy di attendibilità personalizzata). 5. Nella sezione Politica di fiducia personalizzata, inserisci la seguente politica di fiducia personalizzata per questo ruolo: `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "Service": "pods.eks.amazonaws.com" }, "Action": [`	Amministratore AWS

Operazione	Description	Competenze richieste

	<pre>"sts:AssumeRole",
	
	"sts:TagSession"
	]
	}
	]
	}</pre>
	6. Nella pagina Add permissio ns (Aggiungi autorizza zioni), scegli Next (Successi vo). Non aggiungi alcuna politica a questo ruolo.
	7. Per Nome ruolo, inserisci `keda-operator` .
	8. Scegli Crea ruolo.	

Operazione	Description	Competenze richieste
Crea il ruolo IAM per l'applica zione di esempio.	1. Nella console IAM, nel riquadro di navigazione, scegli Ruoli. 2. Scegli Crea ruolo. 3. Scegli il tipo di ruolo Custom trust policy (Policy di attendibilità personali zzata). 4. Nella sezione Politica di fiducia personalizzata, inserisci la seguente politica di fiducia personalizzata per questo ruolo. Sostituis ci `<account number>` con il tuo numero di account di destinazione: `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principa l": { "Service": "pods.eks .amazonaws.com", "AWS": "arn:aws:iam::<acc ount number>:role/ keda-operator" }, "Action": [`	Amministratore AWS

Operazione	Description	Competenze richieste

	<pre>"sts:AssumeRole",
	
	"sts:TagSession"
	]
	}
	]
	}</pre>
	5. Nella pagina Aggiungi autorizzazioni, aggiungi le seguenti politiche AWS gestite al ruolo:
	• `AmazonSQSReadOnlyAccess`
	• `AWSLambdaSQSQueueExecutionRole`
	6. Scegli Next (Successivo).
	7. Per Nome ruolo, inserisci `keda-identity` .
	8. Scegli Crea ruolo.	

Creare una coda Amazon SQS.	1. Apri la console Amazon SQS. 2. Scegliere Crea coda. 3. Per Tipo, scegliere Standard. 4. In Crea pagina di coda, per Nome, inserisci `event-messages-queue` . 5. Scegliere Crea coda. Non modificate nessuna delle impostazioni predefinite per questa coda.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Crea un repository Amazon ECR.	1. Apri la console di Amazon ECR. 2. Scegli Create repository (Crea repository). 3. Per il nome del repository, inserisci `keda-pod-identity-registry` . 4. Scegli Create repository (Crea repository). Non si modifica nessuna delle impostazioni predefinite per questo repository.	Informazioni generali su AWS

Configura il cluster Amazon EKS

Operazione	Description	Competenze richieste
Implementa l'agente Amazon EKS Pod Identity.	Per il cluster Amazon EKS di destinazione, configura l'agente Amazon EKS Pod Identity. Segui le istruzioni in Configurare Amazon EKS Pod Identity Agent nella documentazione di Amazon EKS.	AWS DevOps
Implementa KEDA.	1. Inserisci i seguenti comandi per distribuire KEDA sul cluster Amazon EKS di destinazione: `# Add Helm Repo for Keda`	DevOps ingegnere

Operazione	Description	Competenze richieste

```
helm repo add kedacore
 https://kedacore.g
ithub.io/charts
# Update Helm repo
helm repo update
# Install Keda
helm install keda
 kedacore/keda --
namespace keda --
create-namespace
```

Per ulteriori informazioni, consulta Deploying with Helm nella documentazione KEDA.

2. Dopo una corretta implementazione, nell'output, verifica che siano state create tre distribuzioni per l'operatore KEDA. Di seguito è riportato un esempio di output:

```
NAME
                READY
   UP-TO-DATE
 AVAILABLE   AGE
keda-admission-
webhooks           1/1
     1            1
        89s
keda-operator
             1/1     1
            1
  89s
keda-operator-
metrics-apiserver
```

Operazione	Description	Competenze richieste
	`1/1 1 1 89s`	
Assegna il ruolo IAM all'account del servizio Kubernetes.	Segui le istruzioni in Assegnare un ruolo IAM a un account di servizio Kubernetes nella documentazione di Amazon EKS. Utilizzare i seguenti valori: • Per il ruolo IAM, inserisci. `keda-operator` • Per lo spazio dei nomi Kubernetes, inserisci. `keda` • Per l'account del servizio Kubernetes, inserisci. `keda-operator`	AWS DevOps
Crea uno spazio dei nomi .	Immetti il seguente comando per creare uno spazio `security` dei nomi nel cluster Amazon EKS di destinazione: `kubectl create ns security`	DevOps ingegnere

Distribuzione di un'applicazione di esempio

Operazione	Description	Competenze richieste
Clonare i file dell'applicazione.	Inserisci il seguente comando per clonare l'auto scaling basato sugli eventi utilizzando	DevOps ingegnere

Operazione	Description	Competenze richieste
	EKS Pod Identity e il repository KEDA da: GitHub `git clone https://github.com/aws-samples/event-driven-autoscaling-using-podidentity-and-keda.git`	

Creazione dell'immagine Docker.	1. Inserisci il seguente comando per navigare nel repository clonato:
	`cd event-driven-autoscaling-using-podidentity-and-keda`
	2. Immettete il seguente comando per creare l'immagine Docker per l'applicazione di esempio:
	`docker build -t keda-pod-identity-registry .`
		DevOps ingegnere

Operazione	Description	Competenze richieste

Invia l'immagine Docker ad Amazon ECR.	1. Nel terminale in cui hai creato l'immagine Docker, inserisci il seguente comando per accedere ad Amazon ECR. Sostituisci <AWS_REGION> e <AWS_ACCOUNT_ID> con i valori del tuo AWS ambiente:
	`aws ecr get-login-password \ --region <AWS_REGION> | docker login \ --username AWS \ --password-stdin <AWS_ACCOUNT_ID>.dkr.ecr.<AWS_REGION>.amazonaws.com`
	2. Immettete il seguente comando per etichettare l'immagine. Sostituisci <AWS_REGION> e <AWS_ACCOUNT_ID> con i valori del tuo AWS ambiente:
	`docker tag keda-pod-identity-registry:latest <AWS_ACCOUNT_ID>.dkr.ecr.<AWS_REGION>.amazonaws.com/keda-pod-identity-registry:latest`
		DevOps ingegnere

Operazione	Description	Competenze richieste
	3. Immetti il seguente comando per inviare l'immagine ad Amazon ECR. Sostituisci `<AWS_REGION>` e `<AWS_ACCOUNT_ID>` con i valori del tuo AWS ambiente: `docker push <AWS_ACCOUNT_ID>.dkr.ecr.<AWS_REGION>.amazonaws.com/keda-pod-identity-registry:latest` **Note** Puoi trovare i comandi push accedendo alla pagina del repository Amazon ECR e scegliendo Visualizza comandi push.	

Operazione	Description	Competenze richieste
Implementare un'applicazione di esempio.	1. Nel repository clonato, apri il file deploy.yaml. 2. Sostituisci <AWS_ACCOUNT_ID> e con i valori del tuo ambiente. <AWS_REGION> 3. Salva e chiudi il file deploy.yaml. 4. Immetti il seguente comando per distribuire l'applicazione di esempio sul cluster Amazon EKS di destinazione: `kubectl apply -f deploy.yaml` Questo comando crea un account di distribuzione e servizio nel cluster.	DevOps ingegnere

Operazione	Description	Competenze richieste

Assegna il ruolo IAM all'account del servizio applicativo.	Effettua una delle seguenti operazioni per associare il ruolo `keda-identity` IAM all'account di servizio per l'applicazione di esempio:
	• Segui le istruzioni in Assegnare un ruolo IAM a un account di servizio Kubernetes nella documentazione di Amazon EKS. Utilizzare i seguenti valori:
	  • Per il ruolo IAM, inserisci. `keda-identity`
	  • Per lo spazio dei nomi Kubernetes, inserisci. `security`
	  • Per l'account del servizio Kubernetes, inserisci. `my-sqs-read-msgs`
	• Immettere il comando seguente. AWS CLI Sostituisci `<cluster-name>` con il nome del cluster Amazon EKS di destinazione e `<role-ARN>` sostituiscilo con l'Amazon Resource Name (ARN) del `keda-identity` ruolo:
		DevOps ingegnere

Operazione	Description	Competenze richieste
	`aws eks create-pod-identity-association \ --cluster-name <cluster-name> \ --role-arn <role-ARN> \ --namespace security \ --service-account my-sqs-read-msgs`	

Operazione	Description	Competenze richieste
Implementa `ScaledObject` e. `TriggerAuthentication`	1. Nel repository clonato, apri il file keda.yaml. 2. Sostituiscilo con `{{AWS_ACCOUNT_ID}}` l'ID del tuo obiettivo. Account AWS 3. `{{AWS_REGION}}` Sostituiscilo con il tuo bersaglio Regione AWS. 4. (Facoltativo) Nelle righe 21—24, aggiorna i parametri per la politica di `ScaledObject` scalabilità. Per ulteriori informazioni su questi parametri, vedere quanto segue: • PollingInterval • Periodo di raffreddamento • idleReplicaCount • minReplicaCount 5. Salva e chiudi il file keda.yaml. 6. Immettete il seguente comando per distribuire le risorse and: `ScaledObject TriggerAuthentication` `kubectl -n security apply -f keda.yaml`	DevOps ingegnere

Prova il ridimensionamento automatico

Operazione	Description	Competenze richieste

Invia messaggi alla coda di Amazon SQS.	1. Inserisci il seguente comando per navigare nel repository clonato:
	`cd event-driven-autoscaling-using-podidentity-and-keda`
	2. Inserisci il seguente comando per inviare messaggi di test alla coda di Amazon SQS:
	`python sqs_send_msg.py`
	Lo script sqs_send_msg.py funge da applicazione che genera messaggi per testare l'autoscaling.
	Note
	Se stai usando Python 3, inserisci . `python3 sqs_send_msg.py`
		DevOps ingegnere

Monitora i pod delle applicazioni.	1. In un terminale diverso, inserisci il seguente comando per monitorare i pod:	DevOps ingegnere

Operazione	Description	Competenze richieste

	`kubectl -n security get po`
	2. Per ogni 5 messaggi non letti nella coda Amazon SQS, KEDA aggiunge un pod. Nell'output del comando precedente, conferma che siano stati aggiunti nuovi pod. Di seguito è riportato un esempio di output:
	`kubectl -n security get po`
	`NAME READY STATUS RESTARTS AGE`
	`q-read-797f4c7589-2bj76 1/1 Running 0 2s`
	`q-read-797f4c7589-4zxph 1/1 Running 0 49s`
	`q-read-797f4c7589-cg9dt 1/1 Running 0 18s`
	`q-read-797f4c7589-slc69 1/1 Running 0 33s`
	3. Al termine del test, nel terminale originale, inserisci CTRL + C (Windows) o	

Operazione	Description	Competenze richieste
	CMD + C (macOS). Ciò interrompe lo script python sqs_send_msg.py.	

risoluzione dei problemi

Problema	Soluzione

L'operatore KEDA non è in grado di scalare l'applicazione.	Immettete il seguente comando per controllare i log del ruolo `keda-operator` IAM:
	`kubectl logs -n keda -l app=keda-operator -c keda-operator`
	
	Se è presente un codice di `HTTP 403` risposta, l'applicazione e lo scaler KEDA non dispongono di autorizzazioni sufficienti per accedere alla coda di Amazon SQS. Completa questa procedura:
	1. Controlla le politiche e le istruzioni IAM relative al `keda-identity` ruolo per confermare che l'accesso alla lettura della coda sia concesso.
	2. Convalida la relazione di fiducia tra i ruoli IAM. Di seguito è riportato un esempio:
	`{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": {`

Problema	Soluzione

```
                "Service": "pods.eks
.amazonaws.com"
            },
            "Action": [
                "sts:AssumeRole",
                "sts:TagSession"
            ]
        }
    ]
}
```

Se si `Assume-Role` verifica un errore, un ruolo IAM del nodo Amazon EKS non è in grado di assumere il ruolo IAM per cui è stato definito`TriggerAuthentication` . Completa questa procedura:

1. Inserisci il seguente comando per eliminare il `keda-operator` pod e crearne uno nuovo:

```
kubectl delete pod keda-operator-
<alphenumeric-value> --namespace
 keda
```

2. Immettete il seguente comando per verificare l'identità che il pod assume:

```
kubectl describe pod <keda-operator-
pod-name> --namespace keda
```

3. Quando il pod si riavvia correttamente, conferma che le due variabili seguenti siano state aggiunte alla descrizione del pod:
   - `AWS_CONTAINER_CREDENTIALS_F ULL_URI`

Problema	Soluzione
	• `AWS_CONTAINER_AUTHORIZATION_TOKEN_FILE`

Risorse correlate

- Configurare Amazon EKS Pod Identity Agent (documentazione Amazon EKS)
- Implementazione di KEDA (documentazione KEDA)
- ScaledObject specifiche (documentazione KEDA)
- Autenticazione con TriggerAuthentication (documentazione KEDA)

Semplifica le implementazioni di PostgreSQL su Amazon EKS utilizzando PGO

Shalaka Dengale, Amazon Web Services

Riepilogo

Questo modello integra Postgres Operator di Crunchy Data (PGO) con Amazon Elastic Kubernetes Service (Amazon EKS) per semplificare le implementazioni di PostgreSQL in ambienti nativi del cloud. PGO offre automazione e scalabilità per la gestione dei database PostgreSQL in Kubernetes. Combinando PGO con Amazon EKS, si forma una solida piattaforma per la distribuzione, la gestione e la scalabilità efficiente dei database PostgreSQL.

Questa integrazione offre i seguenti vantaggi chiave:

- Distribuzione automatizzata: semplifica l'implementazione e la gestione dei cluster PostgreSQL.
- Definizioni di risorse personalizzate (CRDs): utilizza le primitive Kubernetes per la gestione di PostgreSQL.
- Alta disponibilità: supporta il failover automatico e la replica sincrona.
- Backup e ripristini automatizzati: semplifica i processi di backup e ripristino.
- Scalabilità orizzontale: consente il ridimensionamento dinamico dei cluster PostgreSQL.
- Aggiornamenti di versione: facilita gli aggiornamenti continui con tempi di inattività minimi.
- Sicurezza: applica la crittografia, i controlli di accesso e i meccanismi di autenticazione.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS.
- AWS Command Line Interface (AWS CLI) versione 2, installata e configurata su Linux, macOS o Windows.
- AWS CLI Config, per connettere AWS le risorse dalla riga di comando.
- eksctl, installato e configurato su Linux, macOS o Windows.
- `kubectl`, installato e configurato per accedere alle risorse sul tuo cluster Amazon EKS. Per ulteriori informazioni, consulta Configurare kubectl ed eksctl nella documentazione di Amazon EKS.

- Il terminale del computer è configurato per accedere al cluster Amazon EKS. Per ulteriori informazioni, consulta Configurare il computer per comunicare con il cluster nella documentazione di Amazon EKS.

Versioni del prodotto

- Versioni di Kubernetes 1.21—1.24 o successive (consulta la documentazione di PGO).
- PostgreSQL versione 10 o successiva. Questo modello utilizza PostgreSQL versione 16.

Limitazioni

- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Architecture

Stack tecnologico Target

- Amazon EKS
- Amazon Virtual Private Cloud (Amazon VPC) (Amazon VPC)
- Amazon Elastic Compute Cloud (Amazon EC2)

Architettura Target

Questo modello crea un'architettura che contiene un cluster Amazon EKS con tre nodi. Ogni nodo viene eseguito su un set di EC2 istanze nel backend. Questa configurazione di PostgreSQL segue un'architettura di replica primaria, particolarmente efficace per i casi d'uso ad alta intensità di lettura. L'architettura include i seguenti componenti:

- Il contenitore del database primario (pg-primary) ospita l'istanza PostgreSQL principale in cui sono dirette tutte le operazioni di scrittura.
- I contenitori di replica secondari (pg-replica) ospitano le istanze PostgreSQL che replicano i dati dal database primario e gestiscono le operazioni di lettura.

- PgBouncerè un pool di connessioni leggero per database PostgreSQL incluso in PGO. Si colloca tra il client e il server PostgreSQL e funge da intermediario per le connessioni al database.
- PGO automatizza l'implementazione e la gestione dei cluster PostgreSQL in questo ambiente Kubernetes.
- Patroni è uno strumento open source che gestisce e automatizza le configurazioni ad alta disponibilità per PostgreSQL. È incluso in PGO. Quando usi Patroni con PGO in Kubernetes, svolge un ruolo cruciale nel garantire la resilienza e la tolleranza ai guasti di un cluster PostgreSQL. Per ulteriori informazioni, consulta la documentazione di Patroni.

Il flusso di lavoro include i seguenti passaggi:

- Implementa l'operatore PGO. Implementa l'operatore PGO sul tuo cluster Kubernetes eseguito su Amazon EKS. Questo può essere fatto utilizzando i manifesti di Kubernetes o i grafici Helm. Questo modello utilizza i manifesti di Kubernetes.
- Definisci le istanze PostgreSQL. Quando l'operatore è in esecuzione, si creano risorse personalizzate (CRs) per specificare lo stato desiderato delle istanze PostgreSQL. Ciò include configurazioni come archiviazione, replica e impostazioni di alta disponibilità.
- Gestione degli operatori. Interagisci con l'operatore tramite oggetti API Kubernetes, ad esempio CRs per creare, aggiornare o eliminare istanze PostgreSQL.
- Monitoraggio e manutenzione. Puoi monitorare lo stato e le prestazioni delle istanze PostgreSQL in esecuzione su Amazon EKS. Gli operatori spesso forniscono metriche e registrazioni per scopi di monitoraggio. Se necessario, è possibile eseguire attività di manutenzione di routine come aggiornamenti e patch. Per ulteriori informazioni, consulta Monitoraggio delle prestazioni del cluster e visualizzazione dei log nella documentazione di Amazon EKS.
- Scalabilità e backup: è possibile utilizzare le funzionalità fornite dall'operatore per ridimensionare le istanze PostgreSQL e gestire i backup.

Questo modello non copre le operazioni di monitoraggio, manutenzione e backup.

Automazione e scalabilità

- È possibile utilizzare CloudFormation per automatizzare la creazione dell'infrastruttura. Per ulteriori informazioni, consulta Creare risorse Amazon EKS CloudFormation nella documentazione di Amazon EKS.
- Puoi utilizzare GitVersion o Jenkins build Numbers per automatizzare la distribuzione delle istanze di database.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire AWS Kubernetes senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire tramite comandi nella shell della riga di comando. Servizi AWS

Altri strumenti

- eksctl è un semplice strumento da riga di comando per la creazione di cluster su Amazon EKS.
- kubectl è un'utilità da riga di comando per eseguire comandi su cluster Kubernetes.
- PGO automatizza e ridimensiona la gestione dei database PostgreSQL in Kubernetes.

Best practice

Segui queste best practice per garantire un'implementazione fluida ed efficiente:

- Proteggi il tuo cluster EKS. Implementa le migliori pratiche di sicurezza per il tuo cluster EKS, come l'utilizzo di ruoli AWS Identity and Access Management (IAM) per account di servizio (IRSA), policy di rete e gruppi di sicurezza VPC. Limita l'accesso al server API del cluster EKS e crittografa le comunicazioni tra i nodi e il server API utilizzando TLS.
- Garantisci la compatibilità delle versioni tra PGO e Kubernetes in esecuzione su Amazon EKS. Alcune funzionalità di PGO potrebbero richiedere versioni specifiche di Kubernetes o introdurre limitazioni di compatibilità. Per ulteriori informazioni, consulta Componenti e compatibilità nella documentazione di PGO.
- Pianifica l'allocazione delle risorse per la tua implementazione PGO, tra cui CPU, memoria e archiviazione. Considera i requisiti di risorse sia di PGO che delle istanze PostgreSQL che gestisce. Monitora l'utilizzo delle risorse e ridimensionale in base alle esigenze.
- Progettato per un'elevata disponibilità. Progetta la tua implementazione PGO per un'elevata disponibilità per ridurre al minimo i tempi di inattività e garantire l'affidabilità. Implementa più repliche di PGO su più zone di disponibilità per la tolleranza agli errori.
- Implementa procedure di backup e ripristino per i database PostgreSQL gestiti da PGO. Utilizza le funzionalità fornite da PGO o soluzioni di backup di terze parti compatibili con Kubernetes e Amazon EKS.

- Configura il monitoraggio e la registrazione per la tua implementazione PGO per tenere traccia delle prestazioni, dello stato e degli eventi. Utilizza strumenti come Prometheus per il monitoraggio delle metriche e Grafana per la visualizzazione. Configura la registrazione per acquisire i registri PGO per la risoluzione dei problemi e il controllo.
- Configura correttamente la rete per consentire le comunicazioni tra PGO, istanze PostgreSQL e altri servizi nel tuo cluster Kubernetes. Utilizza le funzionalità di rete Amazon VPC e i plug-in di rete Kubernetes come Calico o Amazon VPC CNI per l'applicazione delle politiche di rete e l'isolamento del traffico.
- Scegli le opzioni di storage appropriate per i tuoi database PostgreSQL, considerando fattori quali prestazioni, durabilità e scalabilità. Usa volumi Amazon Elastic Block Store (Amazon EBS) AWS o servizi di storage gestito per lo storage persistente. Per ulteriori informazioni, consulta Archivia volumi Kubernetes con Amazon EBS nella documentazione di Amazon EKS.
- Utilizza strumenti Infrastructure as Code (IaC) CloudFormation per automatizzare la distribuzione e la configurazione di PGO su Amazon EKS. Definisci i componenti dell'infrastruttura, tra cui il cluster EKS, il networking e le risorse PGO, come codice per garantire coerenza, ripetibilità e controllo delle versioni.

Epiche

Creazione di un ruolo IAM

Operazione	Description	Competenze richieste
Crea un ruolo IAM.	1. Crea un ruolo IAM utilizzando il seguente comando in AWS CLI: `aws iam create-role \ --role-name {YourRoleName} \ --assume-role-policy-document '{ "Version": "2012-10-17", "Statement": [ {`	Amministratore AWS

Operazione	Description	Competenze richieste

	<pre>"Effect": "Allow",
	
	"Principal": {
	
	"Service": "eks.amazonaws.com"
	},
	
	"Action": "sts:AssumeRole"
	}
	]
	}' && \
	aws iam attach-role-policy \
	--role-name {YourRoleName}\
	--policy-arn arn:aws:iam::aws:policy/AmazonEKSClusterPolicy && \
	aws iam attach-role-policy \
	--role-name {YourRoleName}\
	--policy-arn arn:aws:iam::aws:policy/AmazonEKSServicePolicy && \
	aws iam attach-role-policy \
	--role-name {YourRoleName}\
	--policy-arn arn:aws:iam::aws:policy/CloudWatchFullAccess</pre> 2. Esamina il ruolo in Console di gestione AWS:	

Operazione	Description	Competenze richieste

	a. Apri la console IAM.
	b. Scegli Ruoli e cerca il nome del ruolo che hai creato.
	c. Verifica che siano allegate le seguenti politiche:
	`AmazonEKS ClusterPolicy`
	`AmazonEKS ServicePolicy`
	`CloudWatc hFullAccess`	

Crea un cluster Amazon EKS.

Operazione	Description	Competenze richieste
Crea un cluster Amazon EKS.	Se hai già distribuito un cluster, salta questo passaggio. Altrimenti, distribui sci un cluster Amazon EKS nel tuo ambiente attuale Account AWS utilizzando `eksctl` Terraform o. CloudFormation Questo modello viene utilizzat o `eksctl` per la distribuzione in cluster.	Amministratore AWS, amministratore Terraform o eksctl, amministratore Kubernetes

Operazione	Description	Competenze richieste

	Note
	Questo modello utilizza Amazon EC2 come gruppo di nodi per Amazon EKS. Se desideri utilizzarlo AWS Fargate, consulta la `managedNodeGroups` configurazione nella documentazione di eksctl.
	
	1. Usa il seguente file `eksctl` di input per generare il cluster.
	
	`sample-cluster.yaml`:
	
	`apiVersion: eksctl.io/v1alpha5`
	`kind: ClusterConfig`
	`metadata:`
	`  name: postgresql`
	`  region: us-east-1`
	`  version: "1.29"`
	`accessConfig:`
	`  authenticationMode: API_AND_CONFIG_MAP`
	`availabilityZones:`
	`- us-east-1a`
	`- us-east-1b`
	`- us-east-1c`
	`nodeGroups:`	

Operazione	Description	Competenze richieste

	<pre> - name: ng-1
	    instanceType:
	 m5.16xlarge
	    desiredCapacity:
	 2
	  - name: ng-2
	    instanceType:
	 m5.16xlarge
	    desiredCapacity:
	 2
	  - name: ng-3
	    instanceType:
	 m5.16xlarge
	    desiredCapacity:
	 2
	vpc:
	  cidr: 192.168.0
	.0/16
	  clusterEndpoints:
	    publicAccess:
	 true
	  nat:
	    gateway: HighlyAva
	ilable
	iamIdentity
	Mappings:
	  - arn: arn:aws:i
	am::<account-id>:r
	ole/<role-name> #
	 update the IAM role
	 ARN created in step 1
	    username: <user-
	name> # Enter the
	 user name per your
	 choice
	    noDuplicateARNs:
	 false</pre>
	2. Esegui il comando seguente per creare il cluster (fornisci il percorso	

Operazione	Description	Competenze richieste
	`sample-cluster.yaml` del file): `eksctl create cluster -f sample-cluster.yaml`	
Convalida lo stato del cluster.	Esegui il comando seguente per visualizzare lo stato corrente dei nodi nel cluster: `kubectl get nodes` Se riscontri errori, consulta la sezione sulla risoluzione dei problemi della documentazione di Amazon EKS.	Amministratore AWS, amministratore Terraform o eksctl, amministratore Kubernetes

Crea un provider di identità OIDC

Operazione	Description	Competenze richieste
Abilita il provider IAM OIDC.	Come prerequisito per il driver Amazon EBS Container Storage Interface (CSI), devi disporre di un provider IAM OpenID Connect (OIDC) esistente per il tuo cluster. Abilita il provider IAM OIDC utilizzando il seguente comando: `eksctl utils associate -iam-oidc-provider`	Amministratore AWS

Operazione	Description	Competenze richieste
	`--region={region} --cluster={YourClusterNameHere} --approve` Per ulteriori informazioni su questo passaggio, consulta la documentazione di Amazon EKS.	

Operazione	Description	Competenze richieste
Crea un ruolo IAM per il driver CSI di Amazon EBS.	Usa il seguente `eksctl` comando per creare il ruolo IAM per il driver CSI: `eksctl create iamservic eaccount \ --region {RegionName} \ --name ebs-csi-c ontroller-sa \ --namespace kube-syst em \ --cluster {YourClus terNameHere} \ --attach-policy-ar n arn:aws:iam::aws:p olicy/service-role /AmazonEBSCSIDrive rPolicy \ --approve \ --role-only \ --role-name AmazonEKS _EBS_CSI_DriverRole` Se utilizzi unità Amazon EBS crittografate, devi configura re ulteriormente la policy. Per istruzioni, consulta la documentazione del driver Amazon EBS SCI.	Amministratore AWS

Operazione	Description	Competenze richieste
Aggiungi il driver Amazon EBS CSI.	Usa il seguente `eksctl` comando per aggiungere il driver Amazon EBS CSI: `eksctl create addon \ --name aws-ebs-csi-driver \ --cluster <YourClusterName> service-account-role-arn arn:aws:iam::$(aws sts get-caller-identity \ --query Account \ --output text):role/AmazonEKS_EBS_CSI_DriverRole \ --force`	Amministratore AWS

Installa PGO

Operazione	Description	Competenze richieste
Clona il repository PGO.	Clona il repository per PGO GitHub : `git clone https://github.com/CrunchyData/postgres-operator-examples.git`	AWS DevOps
Fornisci i dettagli del ruolo per la creazione dell'account di servizio.	Per concedere al cluster Amazon EKS l'accesso alle AWS risorse richieste, specifica l'Amazon Resource Name (ARN) del ruolo OIDC che hai creato in precedenz	Amministratore AWS, amministratore Kubernetes

Operazione	Description	Competenze richieste

	a nel file. `service_account.yaml` Questo file si trova nella cartella namespace del repository.
	
	`cd postgres-operator-examples`
	
	`---`
	`metadata:`
	`  annotations:`
	`  eks.amazonaws.com/role-arn: arn:aws:iam::<accountId>:role/<role_name> # Update the OIDC role ARN created earlier`	

Operazione	Description	Competenze richieste

Crea il namespace e i prerequisiti PGO.	1. Esegui il comando seguente per creare lo spazio dei nomi .
	`kubectl apply -k kustomize/install/namespace`
	Questo stabilisce uno spazio dei nomi dedicato per PGO. Se necessario, è possibile modificare il `namespace.yml` file e assegnare allo spazio dei nomi un nome diverso.
	2. Esegui il comando seguente per applicare la configurazione predefinita al cluster:
	`kubectl apply --server-side -k kustomize/install/default`
	`kustomize/install/default` fornisce la configurazione predefinita per i file RBAC (Role-Based Access Control) di Kubernetes, Custom Resource Definition (CRD) e Kubernetes Manager.
		Amministratore Kunernetes

Operazione	Description	Competenze richieste
Verifica la creazione dei pod.	Verifica che lo spazio dei nomi e la configurazione predefinita siano stati creati: `kubectl get pods -n postgres-operator`	Amministratore AWS, amministratore Kubernetes
Verifica. PVCs	Utilizzate il seguente comando per verificare le dichiarazioni di volume persistenti (PVCs): `kubectl describe pvc -n postgres-operator`	Amministratore AWS, amministratore Kubernetes

Crea e distribuisci un operatore

Operazione	Description	Competenze richieste

Crea un operatore.	Modifica il contenuto del file che si trova in `/kustomize/postgres/postgres.yaml` modo che corrisponda a quanto segue:
	`spec:`
	`  instances:`
	`  - name: pg-1`
	`  replicas: 3`
	`  patroni:`
	`  dynamicConfiguration:`
	`  postgresql:`
	`  pg_hba:`
	`  - "host all all 0.0.0.0/0 trust"`
	`  # this line enabled`
		Amministratore AWS, DBA, amministratore Kubernetes

Operazione	Description	Competenze richieste

	<pre>logical replication
	with programmatic
	access
	- "host all
	postgres 127.0.0.1/32
	md5"
	synchronous_mode:
	true
	users:
	- name: replicator
	databases:
	- testdb
	options: "REPLICAT
	ION"</pre>
	Questi aggiornamenti eseguono le seguenti operazioni:
	• Modifica le impostazioni di configurazione di PostgreSQL per facilitare l'accesso all'istanza PostgreSQL.
	• Includi configurazioni per utente di replica, utente del database e superutente per abilitare la replica in streaming, l'accesso al database e la gestione dei cluster.	

Operazione	Description	Competenze richieste
Implementa l'operatore.	Implementa l'operatore PGO per consentire la gestione e il funzionamento semplificati dei database PostgreSQL negli ambienti Kubernetes: `kubectl apply -k kustomize/postgres`	Amministratore AWS, DBA, amministratore Kubernetes

Verifica la distribuzione.	1. Verifica che l'operatore sia stato schierato:
	`kubectl get pods -n postgres-operator --selector=postgres-operator.crunchydata.com/instance-set \ -L postgres-operator.crunchydata.com/role`
	2. Verifica che la risorsa di servizio associata al pod dell'operatore sia stata creata:
	`kubectl get svc -n postgres-operator`
	Dall'output del comando, annota la replica principale (`primary_pod_name`) e leggi replica (`read_pod_name`). Li utilizzerai nei passaggi successivi.
		Amministratore AWS, DBA, amministratore Kubernetes

Verifica la replica in streaming

Operazione	Description	Competenze richieste

Scrive i dati sulla replica principale.	Usa i seguenti comandi per connetterti alla replica primaria di PostgreSQL e scrivere dati nel database:
	`kubectl exec -it <primary_pod_name> bash -n postgres-operator`
	`psql`
	`CREATE TABLE customers (firstname text, customer_id serial, date_created timestamp );`
	`\dt`
		Amministratore AWS, amministratore Kubernetes

Verifica che la replica letta contenga gli stessi dati.	Connect alla replica di lettura PostgreSQL e verifica se la replica in streaming funziona correttamente:
	`kubectl exec -it {read_pod_name} bash -n postgres-operator`
	`psql`
	`\dt`
		Amministratore AWS, amministratore Kubernetes

Operazione	Description	Competenze richieste
	La replica di lettura dovrebbe contenere la tabella creata nella replica primaria nel passaggio precedente.	

risoluzione dei problemi

Problema	Soluzione

Il pod non si avvia.	• Usa il seguente comando per controllare lo stato del pod:
	`kubectl get pods -n your-namespace`
	• Ispeziona i log per eventuali errori:
	`kubectl logs your-pod-name -n your-namespace`
	• Controlla gli eventi dei pod per eventuali eventi anomali correlati ai tuoi pod:
	`kubectl describe pod your-pod-name -n your-namespace`

Le repliche sono notevolmente inferiori rispetto al database principale.	• Verifica il ritardo di replica:
	`SELECT * FROM pg_stat_replication;`
	• Assicurati che le repliche abbiano risorse di CPU e memoria sufficienti. Verifica i limiti delle risorse:
	`kubectl describe pod your-replica-pod -n your-namespace`

Problema	Soluzione
	• Verifica che il backend di archiviazione funzioni in modo ottimale. Un disco lento I/O può causare ritardi nella replica.
Non hai visibilità sulle prestazioni e sullo stato del cluster PostgreSQL.	• Abilita Amazon CloudWatch Logs e assicurat i che i log vengano inviati ad Amazon CloudWatch per l'analisi. Per ulteriori informazioni, consulta la documentazione di Amazon EKS. • Controlla **`pg_stat_activity`** : `SELECT * FROM pg_stat_activity;`

Problema	Soluzione

La replica non funziona.	• Controlla la configurazione principale visualizzando le impostazioni di replica in: `postgresql.conf`
	`wal_level = replica`
	`max_wal_senders = 10`
	`wal_keep_size = 64  # or wal_keep_segments in older versions`
	• Verifica che `pg_hba.conf` includa le autorizzazioni di replica:
	`host replication replica_user all md5`
	• Controlla la configurazione della replica. Assicurati che le impostazioni `recovery.conf` o le impostazioni equivalenti (`standby.signal` e `primary_conninfo`) siano configurate correttamente sulle repliche.

Risorse correlate

- Amazon Elastic Kubernetes Service (white paper sulla panoramica delle opzioni di distribuzione su AWS)
- CloudFormation(Panoramica delle opzioni di distribuzione nel white paper di AWS)
- Inizia a usare Amazon EKS — eksctl (Amazon EKS User Guide)
- Configura kubectl ed eksctl (Guida per l'utente di Amazon EKS)
- Creare un ruolo per la federazione OpenID Connect (IAM User Guide)
- Configurazione delle impostazioni per la AWS CLI(Guida per l'AWS CLIutente)

- Documentazione di Crunchy Postgres per Kubernetes
- Crunch & Learn: Crunchy Postgres per Kubernetes 5.0 (video)

Semplifica l'autenticazione delle applicazioni con TLS reciproco in Amazon ECS utilizzando Application Load Balancer

Olawale Olaleye e Shamanth Devagari, Amazon Web Services

Riepilogo

Questo modello ti aiuta a semplificare l'autenticazione delle applicazioni e ad alleggerire gli oneri di sicurezza con il TLS reciproco in Amazon Elastic Container Service (Amazon ECS) utilizzando Application Load Balancer (ALB). Con ALB, puoi autenticare i certificati client X.509 da. AWS Autorità di certificazione privata Questa potente combinazione aiuta a garantire una comunicazione sicura tra i servizi, riducendo la necessità di complessi meccanismi di autenticazione all'interno delle applicazioni. Inoltre, il modello utilizza Amazon Elastic Container Registry (Amazon ECR) per archiviare le immagini dei container.

L'esempio di questo modello utilizza immagini Docker da una galleria pubblica per creare inizialmente i carichi di lavoro di esempio. Successivamente, vengono create nuove immagini Docker per essere archiviate in Amazon ECR. Per la fonte, prendi in considerazione un sistema basato su Git come GitHub, o Bitbucket GitLab, oppure usa Amazon Simple Storage Service Amazon S3 (Amazon S3). Per creare le immagini Docker, valuta la possibilità di utilizzarle per le immagini successive. AWS CodeBuild

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS con accesso agli stack di distribuzione. AWS CloudFormation Assicurati di disporre delle autorizzazioni utente o di ruolo AWS Identity and Access Management (IAM) per la distribuzione. CloudFormation
- AWS Command Line Interface (AWS CLI) installato. Configurate AWS le credenziali sul computer locale o nell'ambiente utilizzando AWS CLI o impostando le variabili di ambiente nel `~/.aws/credentials` file.
- OpenSSL installato.
- Docker installato.
- Familiarità con quanto Servizi AWS descritto in Strumenti.
- Conoscenza di Docker e NGINX.

Limitazioni

- Mutual TLS for Application Load Balancer supporta solo i certificati client X.509v3. I certificati client X.509v1 non sono supportati.
- Il CloudFormation modello fornito nell'archivio di codice di questo pattern non include il provisioning di un CodeBuild progetto come parte dello stack.
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità regionale, vedi AWS Servizi per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- Docker versione 27.3.1 o successiva
- AWS CLI versione 2.14.5 o successiva

Architecture

Il diagramma seguente mostra i componenti dell'architettura di questo modello.

Il diagramma mostra il flusso di lavoro seguente:

1. Crea un repository Git ed esegui il commit del codice dell'applicazione nel repository.
2. Crea un'autorità di certificazione (CA) privata in. AWS Private CA
3. Crea un CodeBuild progetto. CodeBuildproject Viene attivato dalle modifiche di commit, crea l'immagine Docker e pubblica l'immagine creata su Amazon ECR.
4. Copia la catena di certificati e l'ente del certificato dalla CA e carica il pacchetto di certificati su Amazon S3.
5. Crea un trust store con il pacchetto CA che hai caricato su Amazon S3. Associate il trust store ai listener TLS reciproci sull'Application Load Balancer (ALB).
6. Utilizza la CA privata per emettere certificati client per i carichi di lavoro dei container. Crea anche un certificato TLS privato utilizzando. AWS Private CA
7. Importa il certificato TLS privato in AWS Certificate Manager (ACM) e usalo con ALB.
8. Il carico di lavoro del container in `ServiceTwo` utilizza il certificato client emesso per autenticarsi con l'ALB quando comunica con il carico di lavoro del contenitore in. `Service0ne`

9. Il carico di lavoro del container in `Service0ne` utilizza il certificato client emesso per autenticarsi con l'ALB quando comunica con il carico di lavoro del contenitore in. `ServiceTwo`

Automazione e scalabilità

Questo modello può essere completamente automatizzato utilizzando CloudFormation o AWS Cloud Development Kit (AWS CDK) eseguendo operazioni API da un SDK per fornire le AWS risorse.

Puoi utilizzarla AWS CodePipeline per implementare una pipeline di integrazione e distribuzione continua (CI/CD) CodeBuild per automatizzare il processo di creazione di immagini dei container e distribuire nuove versioni ai servizi del cluster Amazon ECS.

Tools (Strumenti)

Servizi AWS

- AWS Certificate Manager (ACM) ti aiuta a creare, archiviare e rinnovare certificati e chiavi SSL/TLS X.509 pubblici e privati che proteggono i tuoi siti Web e le tue applicazioni. AWS
- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita tra e. Account AWS Regioni AWS
- AWS CodeBuildè un servizio di compilazione completamente gestito che consente di compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per l'implementazione.
- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- Amazon Elastic Container Service (Amazon ECS) è un servizio di gestione dei container veloce e altamente scalabile per l'esecuzione, l'arresto e la gestione dei container su un cluster. Puoi eseguire le tue attività e i tuoi servizi su un'infrastruttura serverless gestita da. AWS Fargate In alternativa, per un maggiore controllo sulla tua infrastruttura, puoi eseguire attività e servizi su un cluster di istanze Amazon Elastic Compute Cloud (Amazon EC2) che gestisci.
- Amazon ECS Exec ti consente di interagire direttamente con i container senza dover prima interagire con il sistema operativo del contenitore host, aprire le porte in entrata o gestire le chiavi SSH. Puoi usare ECS Exec per eseguire comandi in o inserire una shell in un contenitore in esecuzione su un' EC2 istanza Amazon o su. AWS Fargate
- Elastic Load Balancing (ELB) distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni. Ad esempio, puoi distribuire il traffico tra EC2 istanze Amazon, contenitori e indirizzi IP, in una o più zone di disponibilità. ELB monitora lo stato dei target registrati e indirizza il traffico solo verso gli obiettivi integri. ELB ridimensiona il sistema di bilanciamento del carico in base alle

variazioni del traffico in entrata nel tempo. Può adattarsi automaticamente alla maggior parte dei carichi di lavoro.

- AWS Fargateti aiuta a eseguire contenitori senza dover gestire server o EC2 istanze Amazon. Fargate è compatibile sia con Amazon ECS che con Amazon Elastic Kubernetes Service (Amazon EKS). Puoi eseguire le tue attività e i tuoi servizi Amazon ECS con il tipo di lancio Fargate o con un provider di capacità Fargate. A tale scopo, impacchetta l'applicazione in contenitori, specifica i requisiti di CPU e memoria, definisci le politiche di rete e IAM e avvia l'applicazione. Ogni attività Fargate ha il proprio limite di isolamento e non condivide il kernel sottostante, le risorse della CPU, le risorse di memoria o l'interfaccia elastica di rete con un'altra attività.
- AWS Autorità di certificazione privataconsente la creazione di gerarchie di autorità di certificazione (CA) private, tra cui root e subordinate CAs, senza i costi di investimento e manutenzione legati alla gestione di una CA locale.

Altri strumenti

- Docker è un insieme di prodotti Platform as a Service (PaaS) che utilizzano la virtualizzazione a livello di sistema operativo per fornire software in container.
- GitHubGitLab, e Bitbucket sono alcuni dei sistemi di controllo del codice sorgente basati su Git comunemente usati per tenere traccia delle modifiche al codice sorgente.
- NGINX Open Source è un sistema open source di bilanciamento del carico, cache dei contenuti e server web. Questo modello lo utilizza come server web.
- OpenSSL è una libreria open source che fornisce servizi utilizzati dalle implementazioni OpenSSL di TLS e CMS.

Archivio di codici

Il codice per questo pattern è disponibile nel repository GitHub MTLS-with-Application-Load-Balancer-in-Amazon-ECS.

Best practice

- Usa Amazon ECS Exec per eseguire comandi o inserire una shell in un contenitore in esecuzione su Fargate. Puoi anche usare ECS Exec per raccogliere informazioni diagnostiche per il debug.
- Utilizza i gruppi di sicurezza e gli elenchi di controllo degli accessi alla rete (ACLs) per controllare il traffico in entrata e in uscita tra i servizi. Le attività Fargate ricevono un indirizzo IP dalla sottorete configurata nel cloud privato virtuale (VPC).

Epiche

Crea il repository

Operazione	Description	Competenze richieste
Scarica il codice sorgente.	Per scaricare il codice sorgente di questo pattern, esegui il fork o clona il repository GitHub MTLS-with-Application-Load-Balancer-in-Amazon-ECS.	DevOps ingegnere
Crea un repository Git.	Per creare un repository Git che contenga il Dockerfile e `buildspec.yaml` i file, usa i seguenti passaggi: 1. Crea una cartella nel tuo ambiente virtuale. Assegna un nome con il nome del tuo progetto. 2. Apri un terminale sul tuo computer locale e accedi a questa cartella. 3. Per clonare il repository MTLS-with-Application-Load-Balancer-in-Amazon-ECS nella directory del tuo progetto, inserisci il seguente comando: `git clone https://github.com/aws-samples/mTLS-with-Application-Load-Balan`	DevOps ingegnere

Operazione	Description	Competenze richieste
	`cer-in-Amazon-ECS.git`	

Crea CA e genera certificati

Operazione	Description	Competenze richieste
Crea una CA privata in AWS Private CA.	Per creare un'autorità di certificazione (CA) privata, esegui i seguenti comandi nel tuo terminale. Sostituite i valori nelle variabili di esempio con i vostri valori. `export AWS_DEFAULT_REGION="us-west-2"` `export SERVICES_DOMAIN="www.example.com"` `export ROOT_CA_ARN=`aws acm-pca create-certificate-authority \` `--certificate-authority-type ROOT \` `--certificate-authority-configuration \` `"KeyAlgorithm=RSA_2048,` `SigningAlgorithm=SHA256WITHRSA,` `Subject={` `Country=US,` `State=WA,` `Locality=Seattle,`	DevOps ingegnere, AWS DevOps

Operazione	Description	Competenze richieste
	`Organizat ion=Build on AWS, Organizat ionalUnit=mTLS Amazon ECS and ALB Example, CommonName= ${SERVICES_DOMAIN}}" \ --query CertificateAuthori tyArn --output text` Per maggiori dettagli, consulta Creare una CA privata AWS Private CA nella AWS documentazione.	

Operazione	Description	Competenze richieste
Crea e installa il tuo certificato CA privato.	Per creare e installare un certificato per la tua CA root privata, esegui i seguenti comandi nel tuo terminale: 1. Genera una richiesta di firma del certificato (CSR). ``` ROOT_CA_CSR=`aws acm-pca get-certificate-authority-csr \     --certificate-authority-arn ${ROOT_CA_ARN} \     --query Csr --output text` ``` 2. Emetti il certificato principale. ``` AWS_CLI_VERSION=$(aws --version 2>&1 | cut -d/ -f2 | cut -d. -f1) [[${AWS_CLI_VERSION} -gt 1]] && ROOT_CA_CSR="$(echo ${ROOT_CA_CSR} | base64)" ROOT_CA_CERT_ARN=`aws acm-pca issue-certificate \ --certificate-authority-arn ${ROOT_CA_ARN} \ --template-arn arn:aws:acm-pca:::template/RootCACertificate/V1 \ ```	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	<pre> --signing- algorithm SHA256WIT HRSA \ --validity Value=10,Type=YEARS \ --csr "${ROOT_C A_CSR}" \ --query Certifica teArn --output text`</pre> 3. Recupera il certificato principale. <pre>ROOT_CA_CERT=`aws acm-pca get-certi ficate \ --certificate-arn ${ROOT_CA_CERT_ARN} \ --certificate- authority-arn ${ROOT_CA_ARN} \ --query Certifica te --output text` # store for later use aws acm-pca get-certi ficate \ --certificate-arn ${ROOT_CA_CERT_ARN} \ --certificate- authority-arn ${ROOT_CA_ARN} \ --query Certifica te --output text > ca-cert.pem</pre>	

Operazione	Description	Competenze richieste

	4. Importa il certificato CA principale per installarlo sulla CA.
	
	`[[ ${AWS_CLI_VERSION} -gt 1 ]] && ROOT_CA_CERT="$(echo ${ROOT_CA_CERT} | base64)"`
	
	`aws acm-pca import-certificate-authority-certificate \ --certificate-authority-arn $ROOT_CA_ARN \ --certificate "${ROOT_CA_CERT}"`
	
	Per ulteriori dettagli, consulta Installazione del certificato CA nella AWS documentazione.	

Operazione	Description	Competenze richieste
Richiedi un certificato gestito.	Per richiedere un certifica to privato AWS Certificate Manager da utilizzare con il tuo ALB privato, usa il seguente comando: `export TLS_CERTI FICATE_ARN=`aws acm request-certificate \ --domain-name "*. ${DOMAIN_DOMAIN}" \ --certificate-auth ority-arn ${ROOT_CA _ARN} \ --query Certifica teArn --output text``	DevOps ingegnere, AWS DevOps

Operazione	Description	Competenze richieste

Utilizza la CA privata per emettere un certificato client.	• Per creare una richiesta di firma del certificato (CSR) per i due servizi, usa il seguente AWS CLI comando:
	
	`openssl req -out client_csr1.pem -new -newkey rsa:2048 -nodes -keyout client_private-key 1.pem`
	
	`openssl req -out client_csr2.pem -new -newkey rsa:2048 -nodes -keyout client_private-key 2.pem`
	
	Questo comando restituisce la CSR e la chiave privata per i due servizi.
	
	• Per emettere un certifica to per i servizi, esegui i seguenti comandi per utilizzare la CA privata che hai creato:
	
	`SERVICE_ONE_CERT_A RN=`aws acm-pca issue-certificate \`
		DevOps ingegnere, AWS DevOps

Operazione	Description	Competenze richieste
	`--certificate-authority-arn ${ROOT_CA_ARN} \` `--csr fileb://client_csr1.pem \` `--signing-algorithm "SHA256WITHRSA" \` `--validity Value=5,Type="YEARS" --query CertificateArn --output text` `echo "SERVICE_ONE_CERT_ARN: ${SERVICE_ONE_CERT_ARN}"` `aws acm-pca get-certificate \` `--certificate-authority-arn ${ROOT_CA_ARN} \` `--certificate-arn ${SERVICE_ONE_CERT_ARN} \` `| jq -r '.Certificate' > client_cert1.cert` `SERVICE_TWO_CERT_ARN=`aws acm-pca issue-certificate \` `--certificate-authority-arn ${ROOT_CA_ARN} \` `--csr fileb://client_csr2.pem \` `--signing-algorithm "SHA256WITHRSA" \` `--validity Value=5,Type="YEARS" --query CertificateArn --output text`	

Operazione	Description	Competenze richieste

	``
	echo "SERVICE_TWO_CERT_
	ARN: ${SERVICE
	_TWO_CERT_ARN}"
	
	aws acm-pca get-certi
	ficate \
	    --certificate-auth
	ority-arn ${ROOT_CA
	_ARN} \
	    --certificate-arn
	 ${SERVICE_TWO_CERT
	_ARN} \
	     | jq -r '.Certifi
	cate' > client_ce
	rt2.cert
	``
	
	Per ulteriori informazioni, consulta Emettere certificati privati per entità finali nella AWS documentazione.	

Fornitura di servizi AWS

Operazione	Description	Competenze richieste
Fornitura Servizi AWS con il CloudFormation modello.	Per effettuare il provision ing del cloud privato virtuale (VPC), del cluster Amazon ECS, dei servizi Amazon ECS, Application Load Balancer e Amazon Elastic Container Registry (Amazon ECR), utilizza il modello. CloudForm ation	DevOps ingegnere

Operazione	Description	Competenze richieste

Ottieni variabili.	Verifica di avere un cluster Amazon ECS con due servizi in esecuzione. Per recuperare i dettagli delle risorse e memorizzarli come variabili, usa i seguenti comandi:
	
	`export LoadBalancerDNS=$(aws cloudformation describe-stacks --stack-name ecs-mtls \ --output text \ --query 'Stacks[0].Outputs[?OutputKey==`LoadBalancerDNS`].OutputValue')`
	
	`export ECRRepositoryUri=$(aws cloudformation describe-stacks --stack-name ecs-mtls \ --output text \ --query 'Stacks[0].Outputs[?OutputKey==`ECRRepositoryUri`].OutputValue')`
	
	`export ECRRepositoryServiceOneUri=$(aws cloudformation describe-stacks --stack-name ecs-mtls \ --output text \ --query 'Stacks[0].Outputs[?OutputKey==`ECRRepositoryServiceOneUri`].OutputValue')`
		DevOps ingegnere

Operazione	Description	Competenze richieste

	`export ECRReposi toryServiceTwoUri= $(aws cloudformation describe-stacks -- stack-name ecs-mtls \ --output text \ --query 'Stacks[0 ].Outputs[?OutputK ey==`ECRRepository ServiceTwoUri`].Ou tputValue')`
	
	`export ClusterName= $(aws cloudformation describe-stacks -- stack-name ecs-mtls \ --output text \ --query 'Stacks[0 ].Outputs[?OutputK ey==`ClusterName`] .OutputValue')`
	
	`export BucketName= $(aws cloudformation describe-stacks -- stack-name ecs-mtls \ --output text \ --query 'Stacks[0 ].Outputs[?OutputK ey==`BucketName`]. OutputValue')`
	
	`export Service1L istenerArn=$(aws cloudformation describe-stacks -- stack-name ecs-mtls \ --output text \ --query 'Stacks[0 ].Outputs[?OutputK ey==`Service1Liste`	

Operazione	Description	Competenze richieste
	nerArn`].OutputVal ue')  export Service2L istenerArn=$(aws cloudformation describe-stacks -- stack-name ecs-mtls \ --output text \ --query 'Stacks[0 ].Outputs[?OutputK ey==`Service2Liste nerArn`].OutputVal ue')	

Operazione	Description	Competenze richieste
Crea un CodeBuild progetto.	Per utilizzare un CodeBuild progetto per creare le immagini Docker per i tuoi servizi Amazon ECS, procedi come segue: 1. Accedi a e apri la Console di gestione AWS CodeBuild console all'indirizzo https://console.aws.amazon.com/codesuite/ codebuild/. 2. Crea un nuovo progetto. Per Source, scegli il repository Git che hai creato. Per informazioni sui diversi tipi di integrazione del repository Git, consulta Lavorare con le connessioni nella AWS documentazione. 3. Conferma che la modalità Privileged sia abilitata. Per creare immagini Docker, questa modalità è necessaria. Altrimenti, l'immagine non verrà creata correttamente. 4. Utilizza il `buildspec .yaml` file personalizzato condiviso per ogni servizio. 5. Fornisci i valori per il nome e la descrizione del progetto.	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	Per maggiori dettagli, consulta Creare un progetto di compilazione AWS CodeBuild nella AWS documentazione.	

Operazione	Description	Competenze richieste

Crea le immagini Docker.	È possibile utilizzare CodeBuild per eseguire il processo di creazione dell'immagine. CodeBuild necessita delle autorizzazioni per interagire con Amazon ECR e lavorare con Amazon S3.
	
	Come parte del processo, l'immagine Docker viene creata e inserita nel registro Amazon ECR. Per dettagli sul modello e sul codice, consulta Informazioni aggiuntive.
	
	(Facoltativo) Per creare localmente a scopo di test, utilizzate il seguente comando:
	
	`# login to ECR`
	`aws ecr get-login`
	`-password | docker`
	`login --username`
	`AWS --password-stdin`
	`$ECRRepositoryUri`
	
	`# build image for`
	`service one`
	`cd /service1`
	`aws s3 cp s3://$Buc`
	`ketName/serviceone/`
	`service1/ --recursive`
	`docker build -t`
	`$ECRRepositoryServ`
	`iceOneUri .`
	`docker push $ECRRepos`
	`itoryServiceOneUri`
		DevOps ingegnere

Operazione	Description	Competenze richieste

	`# build image for service two`
	`cd ../service2`
	`aws s3 cp s3://$BucketName/servicetwo/service2/ --recursive`
	`docker build -t $ECRRepositoryServiceTwoUri .`
	`docker push $ECRRepositoryServiceTwoUri`	

Abilita il TLS reciproco

Operazione	Description	Competenze richieste
Carica il certificato CA su Amazon S3.	Per caricare il certificato CA nel bucket Amazon S3, usa il seguente comando di esempio: `aws s3 cp ca-cert.pem s3://$BucketName/acm-trust-store/`	AWS DevOps, DevOps ingegnere
Crea il trust store.	Per creare il trust store, usa il seguente comando di esempio: `TrustStoreArn=`aws elbv2 create-trust-store --name acm-pca-trust-certs \` `--ca-certificates-bundle-s3-bucket $BucketName \`	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	`--ca-certificates-bundle-s3-key acm-trust-store/ca-cert.pem --query 'TrustStores[].TrustStoreArn' --output text`	

Carica i certificati dei clienti.	Per caricare certificati client su Amazon S3 per immagini Docker, usa il seguente comando di esempio:
	`# for service one`
	`aws s3 cp client_cert1.cert s3://$BucketName/serviceone/`
	`aws s3 cp client_private-key1.pem s3://$BucketName/serviceone/`
	
	`# for service two`
	`aws s3 cp client_cert2.cert s3://$BucketName/servicetwo/`
	`aws s3 cp client_private-key2.pem s3://$BucketName/servicetwo/`
		AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste

Modifica l'ascoltatore.	Per abilitare il TLS reciproco sull'ALB, modifica i listener HTTPS utilizzando i seguenti comandi:
	
	`aws elbv2 modify-listener \`
	`  --listener-arn $Service1ListenerArn \`
	`  --certificates CertificateArn=$TLS_CERTIFICATE_ARN_TWO \`
	`  --ssl-policy ELBSecurityPolicy-2016-08 \`
	`  --protocol HTTPS \`
	`  --port 8080 \`
	`  --mutual-authentication Mode=verify,TrustStoreArn=$TrustStoreArn,IgnoreClientCertificateExpiry=false`
	
	`aws elbv2 modify-listener \`
	`  --listener-arn $Service2ListenerArn \`
	`  --certificates CertificateArn=$TLS_CERTIFICATE_ARN_TWO \`
	`  --ssl-policy ELBSecurityPolicy-2016-08 \`
	`  --protocol HTTPS \`
	`  --port 8090 \`
		AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	`--mutual-authentication Mode=verify,TrustStoreArn=$TrustStoreArn,IgnoreClientCertificateExpiry=false` Per ulteriori informazioni, consulta Configurazione del TLS reciproco su un Application Load Balancer nella documentazione. AWS	

Aggiorna i servizi

Operazione	Description	Competenze richieste
Aggiorna la definizione delle attività di Amazon ECS.	Per aggiornare la definizione del task di Amazon ECS, modifica il `image` parametro nella nuova revisione. Per ottenere i valori per i rispettivi servizi, aggiorna le definizioni delle attività con il nuovo Uri delle immagini Docker che hai creato nei passaggi precedenti: oppure `echo $ECRRepositoryServiceOneUri` `echo $ECRRepositoryServiceTwoUri` `"containerDefinitions": [`	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	{ "name": "nginx", "image": "public.ecr.aws/nginx/nginx:latest", # <----- change to new Uri "cpu": 0, Per ulteriori informazioni, consulta la sezione Aggiornamento di una definizione di attività Amazon ECS utilizzando la console nella AWS documentazione.	

Operazione	Description	Competenze richieste
Aggiorna il servizio Amazon ECS.	Aggiorna il servizio con la definizione di attività più recente. Questa definizione di attività è il modello per le immagini Docker di nuova creazione e contiene il certifica to client richiesto per l'autenti cazione TLS reciproca. Per aggiornare il servizio, utilizzare la seguente procedura: 1. Apri la console Amazon ECS nella versione 2https://console.aws.amazon.com/ecs/. 2. Nella pagina Cluster, scegliere il cluster. 3. Nella pagina dei dettagli del cluster, nella sezione Servizi, seleziona la casella di controllo accanto al servizio, quindi scegli Aggiorna. 4. Per fare in modo che il tuo servizio inizi una nuova implementazione, seleziona Force new deployment (Forza una nuova implementazione). 5. Per Definizione delle attività, scegli la famiglia di	Amministratore AWS, AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	definizioni delle attività e la revisione più recente. 6. Scegliere Aggiorna. Ripeti i passaggi per l'altro servizio.	

Accedi all'applicazione

Operazione	Description	Competenze richieste
Copia l'URL dell'applicazione.	Usa la console Amazon ECS per visualizzare l'attività. Quando lo stato dell'attività è stato aggiornato a In esecuzione, seleziona l'attività. Nella sezione Attività, copia l'ID dell'attività.	Amministratore AWS, AWS DevOps

Testa la tua applicazione.	Per testare l'applicazione, usa ECS Exec per accedere alle attività.
	1. Per il primo servizio, usa il seguente comando:
	
	`container="nginx"`
	
	`ECS_EXEC_TASK_ARN ="<TASK ARN>" aws ecs execute-command --cluster $ClusterName \ --task $ECS_EXEC_TASK_ARN \`
		Amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste
	`--container $container \ --interactive \ --command "/bin/bash"`	
	2. Nel contenitore del task del servizio 1, utilizzate il comando seguente per immettere il sistema di bilanciamento del carico interno `url` e la porta del listener che punta al servizio due. Quindi specifica il percorso del certificato del client per testare l'applicazione: `curl -kvs https://<internal-alb-url>:8090 --key /usr/local/share/ca-certificates/client.key --cert /usr/local/share/ca-certificates/client.crt`	
	3. Nel contenitore del task service two, utilizzate il seguente comando per inserire il load balancer interno `url` e la porta del listener che punta al servizio 1. Quindi specifica il percorso del certificato del client per testare l'applicazione:	

Operazione	Description	Competenze richieste
	`curl -kvs https://<internal-alb-url>:8080 --key /usr/local/share/ca-certificates/client.key --cert /usr/local/share/ca-certificates/client.crt` **Note** Il `-k` flag nei comandi curl (come parte di`-kvs`) disabilita la convalida del certificato SSL. Puoi rimuovere questo flag quando usi un certificato SSL che corrisponde al tuo nome di dominio, abilitando la corretta convalida del certificato.	

Risorse correlate

Documentazione Amazon ECS

- Creazione di una definizione di attività Amazon ECS utilizzando la console
- Creazione di un'immagine di container da utilizzare su Amazon ECS
- Cluster Amazon ECS
- Amazon ECS per AWS Fargate

- Best practice di rete di Amazon ECS
- Parametri di definizione del servizio Amazon ECS

Altre risorse AWS

- Come posso usare una CA AWS privata per configurare gli MTL sull'Application Load Balancer? ()AWS re:Post

Informazioni aggiuntive

Modifica del Dockerfile

Il codice seguente mostra i comandi modificati nel Dockerfile per il servizio 1:

```
FROM public.ecr.aws/nginx/nginx:latest
WORKDIR /usr/share/nginx/html
RUN echo "Returning response from Service 1: Ok" > /usr/share/nginx/html/index.html
ADD client_cert1.cert client_private-key1.pem /usr/local/share/ca-certificates/
RUN chmod -R 400 /usr/local/share/ca-certificates/
```

Il codice seguente mostra i comandi modificati nel Dockerfile per il servizio 2:

```
FROM public.ecr.aws/nginx/nginx:latest
WORKDIR /usr/share/nginx/html
RUN echo "Returning response from Service 2: Ok" > /usr/share/nginx/html/index.html
ADD client_cert2.cert client_private-key2.pem /usr/local/share/ca-certificates/
RUN chmod -R 400 /usr/local/share/ca-certificates/
```

Se stai creando le immagini Docker con CodeBuild, il `buildspec` file utilizza il numero di CodeBuild build per identificare in modo univoco le versioni delle immagini come valore del tag. Puoi modificare il `buildspec` file in base alle tue esigenze, come mostrato nel seguente codice `buildspec` personalizzato:

```
version: 0.2

phases:
  pre_build:
    commands:
      - echo Logging in to Amazon ECR...
```

```
      - aws ecr get-login-password --region $AWS_DEFAULT_REGION | docker login --
username AWS --password-stdin $ECR_REPOSITORY_URI
      - COMMIT_HASH=$(echo $CODEBUILD_RESOLVED_SOURCE_VERSION | cut -c 1-7)
      - IMAGE_TAG=${COMMIT_HASH:=latest}
  build:
    commands:
        # change the S3 path depending on the service
      - aws s3 cp s3://$YOUR_S3_BUCKET_NAME/serviceone/ $CodeBuild_SRC_DIR/ --
recursive
      - echo Build started on `date`
      - echo Building the Docker image...
      - docker build -t $ECR_REPOSITORY_URI:latest .
      - docker tag $ECR_REPOSITORY_URI:latest $ECR_REPOSITORY_URI:$IMAGE_TAG
  post_build:
    commands:
      - echo Build completed on `date`
      - echo Pushing the Docker images...
      - docker push $ECR_REPOSITORY_URI:latest
      - docker push $ECR_REPOSITORY_URI:$IMAGE_TAG
      - echo Writing image definitions file...
      # for ECS deployment reference
      - printf '[{"name":"%s","imageUri":"%s"}]' $CONTAINER_NAME $ECR_REPOSITORY_URI:
$IMAGE_TAG > imagedefinitions.json

artifacts:
  files:
    - imagedefinitions.json
```

Altri modelli

- Eliminazione automatica degli AWS CloudFormation stack e delle risorse associate
- Automatizza la gestione dinamica delle pipeline per l'implementazione di soluzioni hotfix in ambienti Gitflow utilizzando e AWS Service Catalog AWS CodePipeline
- Crea automaticamente CI/CD pipeline e cluster Amazon ECS per microservizi utilizzando AWS CDK
- Crea e invia immagini Docker ad Amazon ECR utilizzando GitHub Actions e Terraform
- Containerizza i carichi di lavoro mainframe che sono stati modernizzati da Blu Age
- Crea un parser di log personalizzato per Amazon ECS utilizzando un router di log Firelens
- Implementa sistemi agentici su Amazon Bedrock con il framework CrewAI utilizzando Terraform
- Implementa un ambiente per applicazioni Blu Age containerizzate utilizzando Terraform
- Implementa la logica di preelaborazione in un modello ML in un singolo endpoint utilizzando una pipeline di inferenza in Amazon SageMaker
- Distribuisci carichi di lavoro dalle DevOps pipeline di Azure ai cluster Amazon EKS privati
- Implementa la diagnostica e la risoluzione dei problemi di Kubernetes basate sull'intelligenza artificiale con K8SGPT e l'integrazione con Amazon Bedrock
- Gestisci le blue/green distribuzioni di microservizi su più account e regioni utilizzando i servizi di codice AWS e le chiavi multiregionali AWS KMS
- Gestisci le applicazioni container locali configurando Amazon ECS Anywhere con AWS CDK
- Esegui la migrazione da Oracle WebLogic ad Apache Tomcat (ToMee) su Amazon ECS
- Imposta uno spazio dati minimo praticabile per condividere i dati tra le organizzazioni
- Modernizza le applicazioni ASP.NET Web Forms su AWS
- Monitora i repository Amazon ECR per le autorizzazioni wildcard utilizzando AWS e AWS Config CloudFormation
- Monitora l'attività delle applicazioni utilizzando CloudWatch Logs Insights
- Configura una CI/CD pipeline per carichi di lavoro ibridi su Amazon ECS Anywhere utilizzando AWS CDK e GitLab
- Configura end-to-end la crittografia per le applicazioni su Amazon EKS utilizzando cert-manager e Let's Encrypt
- Semplifica la distribuzione di applicazioni multi-tenant Amazon EKS utilizzando Flux

- Semplifica i flussi di lavoro di machine learning dallo sviluppo locale agli esperimenti scalabili utilizzando SageMaker AI e Hydra
- Struttura un progetto Python in architettura esagonale utilizzando AWS Lambda
- AWS Infrastruttura di test utilizzando LocalStack e Terraform Tests
- Coordina la dipendenza dalle risorse e l'esecuzione delle attività utilizzando il costrutto AWS Fargate WaitCondition hook
- Usa gli agenti Amazon Bedrock per automatizzare la creazione di controlli di accesso in Amazon EKS tramite istruzioni basate su testo

Serverless

Argomenti

- Crea un'app mobile React Native senza server utilizzando AWS Amplify
- Manage tenants across multiple SaaS products on a single control plane (Gestione dei tenant su più prodotti SaaS su un unico piano di controllo (control-plane))
- Consolida la generazione di URL predefiniti di Amazon S3 e il download di oggetti utilizzando un endpoint associato a indirizzi IP statici
- Crea una EventBridge connessione Amazon tra più account in un'organizzazione
- Distribuisci i record DynamoDB ad Amazon S3 utilizzando Kinesis Data Streams e Firehose con AWS CDK
- Implementa il controllo delle versioni delle API basato sul percorso utilizzando domini personalizzati in Amazon API Gateway
- Importa la libreria psycopg2 per interagire con il tuo AWS Lambda database PostgreSQL
- Integra Amazon API Gateway con Amazon SQS per gestire REST asincrono APIs
- Elabora gli eventi in modo asincrono con Amazon API Gateway e AWS Lambda
- Elabora gli eventi in modo asincrono con Amazon API Gateway e Amazon DynamoDB Streams
- Elabora gli eventi in modo asincrono con Amazon API Gateway, Amazon SQS e AWS Fargate
- Esegui le attività di AWS Systems Manager Automation in modo sincrono da AWS Step Functions
- Esegui letture parallele di oggetti S3 usando Python in una funzione AWS Lambda
- Invia dati di telemetria da AWS Lambda a OpenSearch per analisi e visualizzazione in tempo reale
- Configura un router cellulare serverless per un'architettura basata su celle
- Configura l'accesso privato a un bucket Amazon S3 tramite un endpoint VPC
- Risolvi i problemi relativi agli stati AWS Step Functions utilizzando Amazon Bedrock
- Altri modelli

Crea un'app mobile React Native senza server utilizzando AWS Amplify

Deekshitulu Pentakota, Amazon Web Services

Riepilogo

Questo modello mostra come creare un backend serverless per un'app mobile React Native utilizzando AWS Amplify e i seguenti servizi AWS:

- AWS AppSync
- Amazon Cognito
- Amazon DynamoDB

Dopo aver configurato e distribuito il backend dell'app utilizzando Amplify, Amazon Cognito autentica gli utenti dell'app e li autorizza ad accedere all'app. AWS interagisce AppSync quindi con l'app frontend e con una tabella DynamoDB di backend per creare e recuperare dati.

> **Note**
>
> Questo modello utilizza una semplice app "ToDoList" come esempio, ma puoi utilizzare una procedura simile per creare qualsiasi app mobile React Native.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Interfaccia a riga di comando Amplify (Amplify CLI), installata e configurata
- XCode (qualsiasi versione)
- Microsoft Visual Studio (qualsiasi versione, qualsiasi editor di codice, qualsiasi editor di testo)
- Familiarità con Amplify
- Familiarità con Amazon Cognito
- Familiarità con AWS AppSync
- Familiarità con DynamoDB

- Familiarità con Node.js
- Familiarità con npm
- Familiarità con React e React Native
- Familiarità con JavaScript e ECMAScript 6 () ES6
- Familiarità con GraphQL

Architecture

Il diagramma seguente mostra un'architettura di esempio per l'esecuzione del backend di un'app mobile React Native nel cloud AWS:

Il diagramma mostra la seguente architettura:

1. Amazon Cognito autentica gli utenti dell'app e li autorizza ad accedere all'app.
2. Per creare e recuperare dati, AWS AppSync utilizza un'API GraphQL per interagire con l'app frontend e una tabella DynamoDB di backend.

Tools (Strumenti)

Servizi AWS

- AWS Amplify è un set di strumenti e funzionalità appositamente progettati che aiuta gli sviluppatori web e mobili di frontend a creare rapidamente applicazioni complete su AWS.
- AWS AppSync fornisce un'interfaccia GraphQL scalabile che aiuta gli sviluppatori di applicazioni a combinare dati provenienti da più fonti, tra cui Amazon DynamoDB, AWS Lambda e HTTP. APIs
- Amazon Cognito fornisce autenticazione, autorizzazione e gestione degli utenti per app Web e mobili.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.

Codice

Il codice per l'applicazione di esempio utilizzata in questo modello è disponibile nel ios-todo-app repository GitHub aws-amplify-react-native-. Per utilizzare i file di esempio, segui le istruzioni nella sezione Epics di questo pattern.

Epiche

Crea ed esegui la tua app React Native

Operazione	Description	Competenze richieste
Configura un ambiente di sviluppo React Native.	Per istruzioni, consulta Configurazione dell'ambiente di sviluppo nella documenta zione di React Native.	Sviluppatore di app

Crea ed esegui l'app mobile ToDoList React Native in iOS Simulator.	1. Crea una nuova directory di progetto per l'app mobile React Native nel tuo ambiente locale eseguendo il seguente comando in una nuova finestra di terminale:
	`npx react-native init ToDoListA mplify`
	2. Passa alla directory principale del progetto eseguendo il seguente comando:
	`cd ToDoListAmplify`
	3. Esegui l'app eseguendo il seguente comando:
	`npx react-native run-ios`
		Sviluppatore di app

Inizializza un nuovo ambiente di backend per l'app

Operazione	Description	Competenze richieste

Crea i servizi di backend necessari per supportare l'app in Amplify.	1. Nel tuo ambiente locale, esegui il seguente comando dalla directory principale del progetto (): ToDoListAmplify
	`amplify init`
	2. Viene visualizzato un messaggio che richiede di fornire informazioni sull'app. Inserisci le informazioni richieste in base al tuo caso d'uso. Quindi, premere Invio.
	
	Per la configurazione ToDoList dell'app utilizzata in questo modello, applica la seguente configurazione di esempio.
	
	Esempio di impostazioni di configurazione dell'app React Native Amplify
	`? Name: ToDoListAmplify`
	`? Environment: dev`
	`? Default editor: Visual Studio Code`
	`? App type: javascript`
	`? Javascript framework : react-native`
		Sviluppatore di app

Operazione	Description	Competenze richieste
	``` ? Source Directory Path: src ? Distribution Directory Path: / ? Build Command: npm run-script build ? Start Command: npm run-script start ? Select the authentic ation method you want to use: AWS profile ? Please choose the profile you want to use: default ``` Per ulteriori informazioni, consulta Creare un nuovo backend Amplify nella documentazione di Amplify Dev Center. **Note** Il `amplify init` comando fornisce le seguenti risorse utilizzando AWS CloudFormation: • Ruoli AWS Identity and Access Management (IAM) per utenti autenticati e non	

Operazione	Description	Competenze richieste
	autenticati (Auth Role e Unauth Role) • Un bucket Amazon Simple Storage Service (Amazon S3) per la distribuzione (per l'app di esempio di questo pattern, Amplify-meta.json) • Un ambiente di backend in Amplify Hosting	

Aggiungi l'autenticazione Amazon Cognito alla tua app Amplify React Native

Operazione	Description	Competenze richieste
Crea un servizio di autentica zione Amazon Cognito.	1. Nel tuo ambiente locale, esegui il seguente comando dalla directory principale del progetto (ToDoListAmplify): `amplify add auth` 2. Viene visualizzato un messaggio che richiede di fornire informazioni sulle impostazioni di configurazione del servizio di autenticazione. Inserisci le informazioni richieste in base al tuo caso d'uso. Quindi, premere Invio. Per la configurazione ToDoList dell'app utilizzata in questo modello, applica la seguente configurazione di esempio.	Sviluppatore di app

Operazione	Description	Competenze richieste

	Esempio di impostazioni di configurazione del servizio di autenticazione
	
	`? Do you want to use the default authentication and security configuration? \`
	`Default configuration`
	
	`? How do you want users to be able to sign in? \`
	`Username`
	
	`? Do you want to configure advanced settings? \`
	`No, I am done`
	
	Note
	Il `amplify add auth` comando crea le cartelle, i file e i file di dipendenza necessari in una cartella locale (amplify) all'interno della directory principale del progetto. Per la configurazione ToDoList dell'app utilizzata in questo schema, il file aws-exports.js viene creato a questo scopo.	

Operazione	Description	Competenze richieste

Distribuisci il servizio Amazon Cognito nel cloud AWS.	1. Dalla directory principal e del progetto, esegui il seguente comando Amplify CLI:
	`amplify push`
	2. Viene visualizzato un prompt per confermare la distribuzione. Immettere Sì. Quindi, premere Invio.
	Note
	Per vedere i servizi distribuiti nel tuo progetto, vai alla console Amplify eseguendo il seguente comando:
	`amplify console`
		Sviluppatore di app

Operazione	Description	Competenze richieste

Installa le librerie Amplify richieste per React Native e le CocoaPods dipendenze per iOS.	1. Installa le librerie client open source Amplify richieste eseguendo il seguente comando dalla directory principale del progetto:
	`npm install aws-amplify aws-amplify-react-native \ amazon-cognito-identity-js @react-native-community/netinfo \ @react-native-async-storage/async-storage`
	2. Installa le CocoaPods dipendenze richieste per iOS eseguendo il seguente comando:
	`npx pod-install`
		Sviluppatore di app

Operazione	Description	Competenze richieste

Importa e configura il servizio Amplify.	Nel file del punto di ingresso dell'app (ad esempio, App.js), importa e carica il file di configurazione del servizio Amplify inserendo le seguenti righe di codice:
	`import Amplify from 'aws-amplify'`
	`import config from './src/aws-exports'`
	`Amplify.configure(config)`
	Note
	Se ricevi un errore dopo l'importazione del servizio Amplify nel file del punto di ingresso dell'app, interrompi l'app. Quindi, apri XCode e seleziona il ToDoListAmplifyfile.xcworkspace dalla cartella iOS del progetto ed esegui l'app.
		Sviluppatore di app

Operazione	Description	Competenze richieste
Aggiorna il file del punto di ingresso dell'app per utilizzare il componente WithAuthenticator Higher-order (HOC).	**Note** L'`withAuthenticator` HOC fornisce flussi di lavoro di accesso, registrazione e password dimenticata nell'app utilizzando solo poche righe di codice. Per ulteriori informazioni, consulta Opzione 1: utilizzare componenti dell'interfaccia utente predefiniti in Amplify Dev Center. Inoltre, componenti di ordine superiore nella documentazione di React. 1. Nel file del punto di ingresso dell'app (ad esempio, App.js), importa l'`withAuthenticator` HOC inserendo le seguenti righe di codice: `import { withAuthenticator } from 'aws-amplify-react-native'`	Sviluppatore di app

Operazione	Description	Competenze richieste

	2. Esporta l'HOC WithAuthenticator inserendo il seguente codice:
	
	`export default withAuthenticator(App)`
	
	Esempio di codice HOC WithAuthenticator
	
	`import Amplify from 'aws-amplify'`
	`import config from './src/aws-exports'`
	`Amplify.configure(config)`
	`import { withAuthenticator } from 'aws-amplify-react-native';`
	
	`const App = () => {`
	`  return null;`
	`};`
	
	`export default withAuthenticator(App);`
	
	Note
	In iOS Simulator, l'app mostra la schermata di accesso fornita	

Operazione	Description	Competenze richieste
	dal servizio Amazon Cognito.	
Verifica la configurazione del servizio di autenticazione.	In iOS Simulator, procedi come segue: 1. Crea un nuovo account nell'app utilizzando un indirizzo email reale. Un codice di verifica viene quindi inviato all'indirizzo e-mail registrato. 2. Verifica la configurazione dell'account utilizzando il codice che ricevi nell'e-mail di verifica. 3. Inserisci il nome utente e la password che hai creato. Quindi, scegli Accedi. Viene visualizzata una schermata di benvenuto. **Note** Puoi anche aprire la console Amazon Cognito e verificare se un nuovo utente è stato creato o meno nel pool di identità.	Sviluppatore di app

Connect un' AppSync API AWS e un database DynamoDB all'app

Operazione	Description	Competenze richieste
Crea un' AppSync API AWS e un database DynamoDB.	1. Aggiungi un' AppSync API AWS alla tua app ed esegui automaticamente il provisioning di un database DynamoDB eseguendo il seguente comando Amplify CLI dalla directory principale del progetto: `amplify add api` 2. Viene visualizzato un prompt che richiede di fornire informazioni sulle impostazioni di configurazione dell'API e del database. Inserisci le informazioni richieste in base al tuo caso d'uso. Quindi, premere Invio. La CLI Amplify apre il file di schema GraphQL nell'editor di testo. Per la configurazione dell' ToDoList app utilizzata in questo modello, applica la seguente configurazione di esempio. Esempio di impostazioni di configurazione dell'API e del database	Sviluppatore di app

Operazione	Description	Competenze richieste
	? Please select from one of the below mentioned services: \ GraphQL ? Provide API name: todolistamplify ? Choose the default authorization type for the API \ Amazon Cognito User Pool Do you want to use the default authentication and security configuration ? Default configuration How do you want users to be able to sign in? \ Username Do you want to configure advanced settings? \ No, I am done. ? Do you want to configure advanced settings for the GraphQL API \ No, I am done. ? Do you have an annotated GraphQL schema? \ No ? Choose a schema template: \	

Operazione	Description	Competenze richieste

	Single object with fields (e.g., "Todo" with ID, name, description)
	
	? Do you want to edit the schema now? \
	Yes
	
	Esempio di schema GraphQL
	
	type Todo @model {
	id: ID!
	name: String!
	description: String
	}	

Operazione	Description	Competenze richieste
Implementa l' AppSync API AWS.	1. Nella directory principal e del progetto, esegui il seguente comando Amplify CLI: `amplify push` 2. Viene visualizzato un prompt che richiede di fornire ulteriori informazi oni sulle impostazioni di configurazione dell'API e del database. Inserisci le informazioni richieste in base al tuo caso d'uso. Quindi, premere Invio. La tua app può ora interagire con l' AppSync API AWS. Per la configurazione ToDoList dell'app utilizzata in questo modello, applica la seguente configurazione di esempio. Esempio di impostazioni di configurazione AppSync dell'API AWS **Note** La seguente configura zione crea l'API GraphQL in AWS AppSync e una tabella Todo in Dynamo DB.	Sviluppatore di app

Operazione	Description	Competenze richieste
	``` ? Are you sure you want to continue? Yes ? Do you want to generate code for your newly created GraphQL API Yes ? Choose the code generation language target javascript ? Enter the file name pattern of graphql queries, mutations and subscriptions src/ graphql/**/*.js ? Do you want to generate/update all possible GraphQL operations - \ queries, mutations and subscriptions Yes ? Enter maximum statement depth \ [increase from default if your schema is deeply nested] 2 ```	

Operazione	Description	Competenze richieste
Connect il frontend dell'app all' AppSync API AWS.	Per utilizzare l' ToDoList app di esempio fornita in questo modello, copia il codice dal file App.js nel ios-todo-app GitHub repository aws-amplify-react-native-. Quindi, integra il codice di esempio nel tuo ambiente locale. Il codice di esempio fornito nel file App.js del repository esegue le seguenti operazioni: • Mostra il modulo per la creazione di un ToDo elemento con i campi Titolo e Descrizione • Visualizza l'elenco delle cose da fare (titolo e descrizione) • Pubblica e recupera dati utilizzando metodi `aws-amplify`	Sviluppatore di app

Risorse correlate

- AWS Amplify
- Amazon Cognito
- AWS AppSync
- Amazon DynamoDB
- React (documentazione React)

Manage tenants across multiple SaaS products on a single control plane (Gestione dei tenant su più prodotti SaaS su un unico piano di controllo (control-plane))

Ramanna Avancha, Kishan Kavala, Anusha Mandava e Jennifer Pascal, Amazon Web Services

Riepilogo

Questo modello mostra come gestire i cicli di vita dei tenant su più prodotti SaaS (Software as a Service) su un unico piano di controllo nel cloud AWS. L'architettura di riferimento fornita può aiutare le organizzazioni a ridurre l'implementazione di funzionalità ridondanti e condivise nei singoli prodotti SaaS e fornire efficienze di governance su larga scala.

Le grandi aziende possono disporre di più prodotti SaaS in diverse unità aziendali. Questi prodotti spesso devono essere forniti per essere utilizzati da tenant esterni con diversi livelli di abbonamento. Senza una soluzione tenant comune, gli amministratori IT devono dedicare del tempo alla gestione di funzionalità indifferenziate su più SaaS APIs, invece di concentrarsi sullo sviluppo delle funzionalità principali del prodotto.

La soluzione tenant comune fornita in questo modello può aiutare a centralizzare la gestione di molte delle funzionalità condivise dei prodotti SaaS di un'organizzazione, tra cui:

- Sicurezza
- Approvvigionamento per gli inquilini
- Archiviazione dei dati degli inquilini
- Comunicazioni con i tenant
- Gestione del prodotto
- Registrazione e monitoraggio delle metriche

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Conoscenza di Amazon Cognito o di un provider di identità (IdP) di terze parti
- Conoscenza di Amazon API Gateway

- Conoscenza di AWS Lambda
- Conoscenza di Amazon DynamoDB
- Conoscenza di AWS Identity and Access Management (IAM)
- Conoscenza di AWS Step Functions
- Conoscenza di AWS CloudTrail e Amazon CloudWatch
- Conoscenza delle librerie e del codice Python
- Conoscenza del SaaS APIs, compresi i diversi tipi di utenti (organizzazioni, inquilini, amministratori e utenti delle applicazioni), i modelli di abbonamento e i modelli di isolamento dei tenant
- Conoscenza dei requisiti SaaS multiprodotto e degli abbonamenti multi-tenant dell'organizzazione

Limitazioni

- Le integrazioni tra la soluzione tenant comune e i singoli prodotti SaaS non sono coperte da questo modello.
- Questo modello distribuisce il servizio Amazon Cognito solo in una singola regione AWS.

Architecture

Stack tecnologico Target

- Gateway Amazon API
- Amazon Cognito
- AWS CloudTrail
- Amazon CloudWatch
- Amazon DynamoDB
- IAM
- AWS Lambda
- Amazon Simple Storage Service (Amazon S3)
- Amazon Simple Notification Service (Amazon SNS)
- Funzioni AWS Step

Architettura Target

Il diagramma seguente mostra un esempio di flusso di lavoro per la gestione dei cicli di vita dei tenant su più prodotti SaaS su un unico piano di controllo nel cloud AWS.

Il diagramma mostra il flusso di lavoro seguente:

1. Un utente AWS avvia azioni relative al provisioning dei tenant, al provisioning dei prodotti o all'amministrazione effettuando una chiamata a un endpoint API Gateway.
2. L'utente viene autenticato da un token di accesso recuperato da un pool di utenti di Amazon Cognito o da un altro IdP.
3. Le singole attività di provisioning o amministrazione vengono eseguite da funzioni Lambda integrate con gli endpoint API API Gateway.
4. L'amministrazione APIs della soluzione Common Tenant (per tenant, prodotti e utenti) raccoglie tutti i parametri di input, le intestazioni e i token richiesti. Quindi, l'amministrazione APIs richiama le funzioni Lambda associate.
5. Le autorizzazioni IAM sia per l'amministrazione che per APIs le funzioni Lambda sono convalidate dal servizio IAM.
6. Le funzioni Lambda archiviano e recuperano i dati dai cataloghi (per tenant, prodotti e utenti) in DynamoDB e Amazon S3.
7. Dopo la convalida delle autorizzazioni, viene richiamato un flusso di lavoro AWS Step Functions per eseguire un'attività specifica. L'esempio nel diagramma mostra un flusso di lavoro di provisioning dei tenant.
8. Le singole attività del flusso di lavoro AWS Step Functions vengono eseguite in un flusso di lavoro predeterminato (macchina a stati).
9. Tutti i dati essenziali necessari per eseguire la funzione Lambda associata a ciascuna attività del flusso di lavoro vengono recuperati da DynamoDB o Amazon S3. Potrebbe essere necessario effettuare il provisioning di altre risorse AWS utilizzando un CloudFormation modello AWS.
10. Se necessario, il flusso di lavoro invia una richiesta di fornitura di risorse AWS aggiuntive per uno specifico prodotto SaaS all'account AWS di quel prodotto.
11. Quando la richiesta ha esito positivo o negativo, il flusso di lavoro pubblica l'aggiornamento di stato come messaggio su un argomento di Amazon SNS.
12. Amazon SNS è abbonato all'argomento Amazon SNS del flusso di lavoro Step Functions.
13. Amazon SNS invia quindi l'aggiornamento dello stato del flusso di lavoro all'utente AWS.
14. I log delle azioni di ogni servizio AWS, incluso un audit trail delle chiamate API, vengono inviati a CloudWatch. È possibile configurare regole e allarmi specifici CloudWatch per ogni caso d'uso.

15. I log vengono archiviati in bucket Amazon S3 per scopi di controllo.

Automazione e scalabilità

Questo modello utilizza un CloudFormation modello per aiutare ad automatizzare l'implementazione della soluzione tenant comune. Il modello può anche aiutarti a scalare rapidamente verso l'alto o verso il basso le risorse associate.

Per ulteriori informazioni, consulta Working with AWS CloudFormation templates nella AWS CloudFormation User Guide.

Tools (Strumenti)

Servizi AWS

- Amazon API Gateway ti aiuta a creare, pubblicare, gestire, monitorare e proteggere REST, HTTP e WebSocket APIs su qualsiasi scala.
- Amazon Cognito fornisce autenticazione, autorizzazione e gestione degli utenti per app Web e mobili.
- AWS ti CloudTrail aiuta a controllare la governance, la conformità e il rischio operativo del tuo account AWS.
- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue risorse AWS e delle applicazioni che esegui su AWS in tempo reale.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.
- AWS Step Functions è un servizio di orchestrazione serverless che ti aiuta a combinare le funzioni di AWS Lambda e altri servizi AWS per creare applicazioni aziendali critiche.

Best practice

La soluzione in questo modello utilizza un unico piano di controllo per gestire l'onboarding di più tenant e fornire l'accesso a più prodotti SaaS. Il piano di controllo aiuta gli utenti amministrativi a gestire altri quattro piani specifici per funzionalità:

- Piano di sicurezza
- Piano del flusso di lavoro
- Piano di comunicazione
- Piano di registrazione e monitoraggio

Epiche

Configura il piano di sicurezza

Operazione	Description	Competenze richieste
Stabilisci i requisiti per la tua piattaforma SaaS multi-tenant.	Stabilisci requisiti dettagliati per quanto segue: • Tenant • Utenti • Roles • Prodotti SaaS • Sottoscrizioni • Scambi di profili	Architetto cloud, amministratore di sistema AWS
Configura il servizio Amazon Cognito.	Segui le istruzioni in Introduzione ad Amazon Cognito nella Amazon Cognito Developer Guide.	Architetto del cloud
Configura le politiche IAM richieste.	Crea le policy IAM richieste per il tuo caso d'uso. Quindi, associa le policy ai ruoli IAM in Amazon Cognito.	Amministratore cloud, architetto cloud, sicurezza AWS IAM

Operazione	Description	Competenze richieste
	Per ulteriori informazioni, consulta Gestire l'accesso utilizzando le policy e il controllo degli accessi basato sui ruoli nella Amazon Cognito Developer Guide.	
Configura le autorizzazioni API richieste.	Configura le autorizzazioni di accesso all'API Gateway utilizzando i ruoli e le policy IAM e gli autorizzatori Lambda. Per istruzioni, consulta le seguenti sezioni della Amazon API Gateway Developer Guide: • Controlla l'accesso a un'API con autorizzazioni IAM • Usa gli autorizzatori API Gateway Lambda	Amministratore cloud, architetto cloud

Configura il piano dati

Operazione	Description	Competenze richieste
Crea i cataloghi di dati richiesti.	1. Crea tabelle DynamoDB per archiviare i dati per i cataloghi degli utenti. Assicurati di includere gli attributi e i ruoli utente. Inoltre, assicuratevi di eseguire la modellazione dei dati sulle tabelle del	DBA

Operazione	Description	Competenze richieste
	catalogo per mantenere gli attributi obbligatori e facoltativi per ogni utente e ruolo. 2. Crea tabelle DynamoDB per archiviare i dati per i cataloghi di prodotti. Assicurati di modellare i casi d'uso specifici per i tuoi prodotti SaaS. 3. Crea tabelle DynamoDB per archiviare i dati per i cataloghi dei tenant. Assicurati di configurare modelli di abbonamento per tenant, prodotti e licenze per abbonamenti multi-SaaS e tag. Per ulteriori informazioni, consulta Configurazione di DynamoDB nella Amazon DynamoDB Developer Guide.	

Configurare il piano di controllo

Operazione	Description	Competenze richieste
Crea funzioni Lambda e API Gateway APIs per eseguire le attività richieste dal piano di controllo.	Crea funzioni Lambda e API Gateway separati APIs per aggiungere, eliminare e gestire quanto segue: • Utenti	Sviluppatore di app

Operazione	Description	Competenze richieste
	• Tenant • Prodotti Per ulteriori informazioni, consulta Using AWS Lambda with Amazon API Gateway nella AWS Lambda Developer Guide.	

Configura il piano del flusso di lavoro

Operazione	Description	Competenze richieste
Identifica le attività che i flussi di lavoro di AWS Step Functions devono eseguire.	Identifica e documenta i requisiti dettagliati del flusso di lavoro di AWS Step Functions per quanto segue: • Utenti • Tenant • Prodotti ⚠ Important Assicurati che le principali parti interessate approvino i requisiti.	Proprietario dell'app
Crea i flussi di lavoro AWS Step Functions richiesti.	1. Crea i flussi di lavoro richiesti per utenti, tenant e prodotti in AWS Step Functions. Per ulteriori	Sviluppatore di app, Build lead

Operazione	Description	Competenze richieste
	informazioni, consulta la AWS Step Functions Developer Guide. 2. Identifica i meccanismi di gestione dei tentativi e degli errori. Per ulteriori informazioni, consulta Gestione degli errori, nuovi tentativi e aggiunta di avvisi alle macchine a stati Step Function sul blog di AWS. 3. Implementa le fasi del flusso di lavoro utilizzando le funzioni Lambda. Per istruzioni, consulta Creazione di una macchina a stati Step Functions che utilizza Lambda nella AWS Step Functions Developer Guide. 4. Integra qualsiasi servizio esterno con AWS Step Functions secondo necessità. 5. Gestisci lo stato di ogni flusso di lavoro in una tabella DynamoDB e comunica lo stato di ogni flusso di lavoro utilizzando Amazon SNS.	

Configura il piano di comunicazione

Operazione	Description	Competenze richieste
Crea argomenti Amazon SNS.	Crea argomenti Amazon SNS per ricevere notifiche su quanto segue: • Stati del flusso di lavoro • Errori • Tentativi Per ulteriori informazioni, consulta l'argomento Creazione di un SNS nella Amazon SNS Developer Guide.	Proprietario dell'app, Cloud architect
Sottoscrivi gli endpoint a ogni argomento di Amazon SNS.	Per ricevere messaggi pubblicati su un argomento di Amazon SNS, devi sottoscrivere un endpoint per ogni argomento. Per ulteriori informazioni, consulta l'argomento Abbonamento a un argomento Amazon SNS nella Amazon SNS Developer Guide.	Sviluppatore di app, architetto cloud

Configura il piano di registrazione e monitoraggio

Operazione	Description	Competenze richieste
Attiva la registrazione per ogni componente della soluzione tenant comune.	Attiva la registrazione a livello di componente per ogni risorsa nella soluzione tenant comune che hai creato. Per le istruzioni, consulta quanto segue: • Come posso attivare i CloudWatch log per la risoluzione dei problemi relativi all'API o WebSocket all'API REST di API Gateway? (Centro di conoscenza AWS) • Registrazione tramite CloudWatch log (AWS Step Functions Developer Guide) • Registrazione delle funzioni AWS Lambda in Python (AWS Lambda Developer Guide) • Registrazione e monitoraggio in Amazon Cognito (Amazon Cognito Developer Guide) • Monitoraggio con Amazon CloudWatch (Amazon DynamoDB Developer Guide)	Sviluppatore di app, amministratore di sistema AWS, amministratore cloud

Operazione	Description	Competenze richieste
	Note Puoi consolidare i log di ogni risorsa in un account di registrazione centralizzato utilizzando le policy IAM. Per ulteriori informazioni, consulta Registrazione centralizzata e barriere di sicurezza per più account.	

Fornisci e distribuisci la soluzione tenant comune

Operazione	Description	Competenze richieste
Crea CloudFormation modelli.	Automatizza l'implementazione e la manutenzione della soluzione Common Tenant completa e di tutti i suoi componenti utilizzando CloudFormation i modelli. Per ulteriori informazioni, consulta la AWS CloudFormation User Guide.	Sviluppatore di app, DevOps ingegnere, CloudFormation sviluppatore

Risorse correlate

- Controlla l'accesso a un'API REST utilizzando i pool di utenti di Amazon Cognito come autorizzatore (Amazon API Gateway Developer Guide)

- Usa gli autorizzatori API Gateway Lambda (Amazon API Gateway Developer Guide)
- Pool di utenti di Amazon Cognito (Amazon Cognito Developer Guide)
- CloudWatch Console per più account e più regioni (Amazon CloudWatch User Guide)

Consolida la generazione di URL predefiniti di Amazon S3 e il download di oggetti utilizzando un endpoint associato a indirizzi IP statici

Song Jin, Eunhye Jo e Jun Soung Lee, Amazon Web Services

Riepilogo

Questo modello semplifica l'accesso ad Amazon Simple Storage Service (Amazon S3) creando file predefiniti sicuri e personalizzati per il download di oggetti. URLs La soluzione fornisce un singolo endpoint con un dominio unico e indirizzi IP statici. È personalizzato per i clienti che richiedono il consolidamento degli endpoint API e Amazon S3 in un dominio unificato con indirizzi IP statici. Il caso d'uso prevede che gli utenti seguano una politica firewall con elenchi di indirizzi IP e domini consentiti, limitando l'accesso delle API a domini e indirizzi IP specifici.

L'architettura utilizza chiavi Servizi AWS, tra cui AWS Global Accelerator Amazon API Gateway AWS Lambda, Application Load Balancer AWS PrivateLink e Amazon S3. Questo design centralizza l'API per la generazione di endpoint predefiniti URLs e Amazon S3 in un unico dominio, collegato a un acceleratore con due indirizzi IP statici. Di conseguenza, gli utenti possono richiedere URLs e scaricare facilmente oggetti Amazon S3 predefiniti tramite un endpoint di dominio unificato con indirizzi IP statici.

Questa architettura è particolarmente vantaggiosa per i clienti con politiche o requisiti di conformità rigorosi, come quelli dei settori pubblico, medico e finanziario.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Una zona pubblica ospitata per il tuo nome di dominio personalizzato
- Un dominio importato in AWS Certificate Manager (ACM) in un formato Regione AWS a tua scelta

Limitazioni

- Il nome del bucket Amazon S3 deve corrispondere al nome di dominio dell'endpoint. Questo requisito serve a garantire che l'endpoint Amazon S3 possa essere servito tramite un unico endpoint API.

- Il nome di dominio personalizzato utilizzato in API Gateway deve essere allineato al nome di dominio del singolo endpoint API.
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità regionale, vedi AWS Servizi per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Architecture

Il diagramma seguente mostra l'architettura e il flusso di lavoro di destinazione per questo modello.

Il diagramma illustra il concetto e il flusso di lavoro seguenti:

1. Un utente avvia una richiesta per generare un URL predefinito utilizzando l'endpoint personalizzato fornito tramite AWS Global Accelerator, utilizzando il nome di dominio personalizzato e gli indirizzi IP associati.
2. Una funzione Lambda genera l'URL predefinito, che punta all'endpoint personalizzato. Risponde con un reindirizzamento 301 che contiene l'URL prefirmato generato. Tramite l'URL prefirmato reindirizzato, l'utente scarica automaticamente l'oggetto utilizzando l'endpoint personalizzato fornito tramite Global Accelerator.

I componenti dell'architettura generale per la generazione di URL predefiniti e il flusso di lavoro per il download degli oggetti sono i seguenti:

- Fornitura di indirizzi IP statici da parte di Global Accelerator.
- Registrazione dell'alias dell'acceleratore come record A nella zona ospitata pubblica di Amazon Route 53 con il nome di dominio personalizzato.
- Creazione di un bucket Amazon S3 con un nome di bucket che corrisponda al nome di dominio personalizzato registrato.
- Creazione di endpoint VPC per API Gateway e il servizio Amazon S3.
- Configurazione di un Application Load Balancer con interfaccia interna per la connessione a Global Accelerator.
- Assegnazione di un nome di dominio personalizzato per API Gateway con un certificato ACM allegato.
- Implementazione di un API Gateway privato integrato con una funzione Lambda.

- La funzione Lambda è dotata di un ruolo AWS Identity and Access Management (IAM) associato (con GetObjectautorizzazioni).

Tools (Strumenti)

Servizi AWS

- Amazon API Gateway ti aiuta a creare, pubblicare, gestire, monitorare e proteggere REST, HTTP e WebSocket APIs su qualsiasi scala.
- Gli Application Load Balancer distribuiscono il traffico delle applicazioni in entrata su più destinazioni, come le istanze Amazon Elastic Compute Cloud (Amazon EC2), in più zone di disponibilità.
- AWS Certificate Manager (ACM) ti aiuta a creare, archiviare e rinnovare certificati e chiavi SSL/TLS X.509 pubblici e privati che proteggono i tuoi siti Web e le tue applicazioni. AWS
- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che consente di definire e fornire Cloud AWS l'infrastruttura in codice.
- AWS Global Acceleratorè un servizio globale che supporta endpoint multipli Regioni AWS. È possibile creare acceleratori che indirizzano il traffico verso endpoint ottimali sulla AWS rete globale. Questo migliora la disponibilità e le prestazioni delle applicazioni Internet utilizzate da un pubblico globale.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS PrivateLinkti aiuta a creare connessioni private unidirezionali dai tuoi cloud privati virtuali (VPCs) a servizi esterni al VPC.
- Amazon Route 53 è un servizio Web DNS altamente scalabile e disponibile.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri strumenti

- Terraform è uno strumento di infrastruttura come codice (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.

Archivio di codici

Puoi implementare questo modello utilizzando AWS CDK o Terraform in base alle tue preferenze. La sezione Epics contiene istruzioni per entrambi i metodi di distribuzione. Il codice per questo pattern è disponibile nei seguenti GitHub repository:

- AWS CDK— s3 - -cdk presignedurl-staticips-endpoint-with
- Terraform — s3- -terraform presignedurl-staticips-endpoint-with

Best practice

- Per migliorare la sicurezza nell'ambiente di produzione, è fondamentale implementare meccanismi di autorizzazione, come Amazon Cognito, per limitare l'accesso all'API di `PresignedUrl` generazione.
- Segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.

Epiche

Prepara l'ambiente

Operazione	Description	Competenze richieste
Scegli un nome di dominio.	Scegli un nome di dominio pubblico per l'endpoint Amazon S3 unificato. Il nome di dominio viene utilizzato anche come nome del bucket Amazon S3.	Amministratore AWS, amministratore di rete
Crea una zona ospitata pubblica	Crea una zona pubblica ospitata in Amazon Route 53. Il nome di dominio deve corrispondere al nome di dominio utilizzato in API Gateway.	Amministratore AWS, amministratore di rete

Operazione	Description	Competenze richieste
Prepara un certificato SSL.	Utilizza AWS Certificate Manager (ACM) per richiedere o importare un certificato SSL per il dominio della tua applicazione web.	Amministratore AWS, amministratore di rete

Implementa il pattern con Terraform

Operazione	Description	Competenze richieste
Configura l'ambiente di sviluppo Terraform.	Per configurare l'ambiente di sviluppo, procedi come segue: 1. Installa Terraform versione 1.0 o successiva. 2. Clona il repository GitHub s3- presignedurl-staticips-endpoint-with -terraform eseguendo il seguente comando in una finestra di terminale: `git clone https://github.com/aws-samples/s3-presignedurl-staticips-endpoint-with-terraform.git`	Amministratore AWS, amministratore cloud
Modifica i **`provider.tf`** file `.tfvars` and.	1. Crea i **`.tfvars`** file richiesti: crea `apg.tfvars` file in entrambe le `2.apigw_s3_lambda` directory `1.vpc_alb_ga` e. Questi file conterranno i valori delle	Amministratore AWS, amministratore cloud

Operazione	Description	Competenze richieste

	variabili specifiche dell'ambiente.
	• Per `1.vpc_alb_ga/apg.tfvars` , crea un file con il seguente modello:
	`hcl`
	`environment_name = "test"`
	`domain = "your-domain.com"`
	`s3_bucket_prefix = "your-bucket-prefix"`
	`aws_region = "ap-northeast-2"`
	`# Change to your preferred region`
	• Per `2.apigw_s3_lambda/apg.tfvars` esempio, crea un file con il seguente modello (dovrai aggiornarlo `vpc_id` dopo aver completato il passaggio 1):
	`hcl`
	`vpc_id = "vpc-xxxxxxxxx"`
	`# Update this with the VPC ID output from step 1`
	`environment_name = "test"`	

Operazione	Description	Competenze richieste
	``` domain = "your-domain.com" s3_bucket_prefix = "your-bucket-prefix" aws_region = "ap-northeast-2" # Change to your preferred region ```	

	2. Modifica i **`provider.tf`** file: in entrambe le `2.apigw_s3_lambda` cartelle `1.vpc_alb_ga` e, modifica i `provider.tf` file in modo che corrispondano alla AWS configurazione locale:
	``hcl
	provider "aws" {
	region = "ap-northeast-2" # Change to your preferred region
	profile = "apg" # Change to your AWS profile name
	}
	``
	Tieni presente quanto segue:
	• Sostituisci i valori segnaposto nei `.tfvars` file con i tuoi valori effettivi.
	• Dopo aver completato la prima implementazione di Terraform	

Operazione	Description	Competenze richieste
	(`1.vpc_alb_ga`), aggiorna il `vpc_id` valore `2.apigw_s3_lambda/apg.tfvars` con l'ID VPC dall'output. • Assicurati che il nome AWS del tuo profilo nel `provider.tf` file corrisponda a un profilo esistente nel file`~/.aws/credentials` . • Il valore del dominio deve corrispondere al dominio della zona ospitata di Route 53. • Il dominio `s3_bucket_prefix` combinato con formerà il nome del bucket Amazon S3 (ad esempio,) . `your-bucket-prefix.your-domain.com` • In caso di modifica`aws_region` , potrebbe essere necessario aggiornare anche le zone di disponibilità codificate nei `main.tf` file in modo che corrispondano a quelle scelte. Regione AWS	

Operazione	Description	Competenze richieste

Fornisci risorse di rete.	Per effettuare il provisioning delle risorse di rete, esegui i seguenti comandi:
	`cd ./2.vpc_alb_ga`
	`terraform init`
	`terraform plan --var-file=apg.tfvars`
	`terraform apply --var-file=apg.tfvars`
	Durante l'esecuzione del `apply` comando, digitate yes quando richiesto.
		Amministratore AWS, amministratore cloud

Esegui il provisioning di API Gateway, Amazon S3 e Lambda.	Per effettuare il provisioning delle risorse di rete, utilizza i seguenti comandi:
	`cd ./2.apigw_s3_lambda`
	`terraform init`
	`terraform plan --var-file=apg.tfvars`
	`terraform apply --var-file=apg.tfvars`
		Amministratore AWS, amministratore cloud

Implementa il pattern con AWS CDK

Operazione	Description	Competenze richieste
Configura l'ambiente di AWS CDK sviluppo.	Per configurare l'ambiente di sviluppo, procedi come segue: 1. Installa il AWS CDK. 2. Clona il repository GitHub s3- presignedurl-stati	Amministratore AWS, amministratore cloud

Operazione	Description	Competenze richieste
	cips-endpoint-with -cdk eseguendo il seguente comando in una finestra di terminale: `git clone https://github.com/aws-samples/s3-presignedurl-staticips-endpoint-with-cdk.git`	

Operazione	Description	Competenze richieste
Configura le impostazioni del dominio nel `config/index.ts` file.	Per modificare le opzioni della variabile costante, utilizzate i seguenti comandi: `export const options = { certificateArn: '{arn of the acm which created before}', dnsAttr: { zoneName: '{public hosted zone name}', hostedZoneId: 'hosted zone Id', }, domainNamePrefix: '{Prefix for the domain}', presignPath: 'presign', objectsPath: 'objects', };` Nei comandi, sostituisci ogni segnaposto con le tue informazioni: • Sostituisci `'{arn of the acm which created before}'` con il tuo certificato ACM Amazon Resource Name (ARN). • Sostituiscilo `'{public hosted zone name}'` con il tuo nome di dominio.	Amministratore AWS, amministratore cloud

Operazione	Description	Competenze richieste
	• `'hosted zone Id'` Sostituiscilo con l'ID della tua zona ospitata su Route 53. • Sostituiscilo `'{Prefix for the domain}'` con il prefisso di dominio che desideri utilizzare. • Facoltativamente, puoi modificare `'objects'` i percorsi `'presign'` e, se necessario, per il tuo caso d'uso specifico.	

Distribuisci gli stack.	Per distribuire due stack, uno per il cloud privato virtuale (VPC) e l'altro per l'applicazione, usa il seguente comando:
	`$ npm install`
	`$ cdk synth`
	`$ cdk deploy --all`
		Amministratore AWS, amministratore cloud

Prova il modello

Operazione	Description	Competenze richieste
Verifica gli indirizzi IP dell'endpoint.	Per verificare che il dominio per questo pattern abbia indirizzi IP statici, utilizzate il seguente comando:	Amministratore di rete

Operazione	Description	Competenze richieste
	`nslookup ${s3-bucket-prefix}.${domain}`	
Carica un file di test che potrai scaricare in seguito.	Carica il file di test nella `'/objects'` cartella del bucket Amazon S3.	Amministratore AWS, amministratore cloud
Richiama l'API per generare un URL predefinito.	Per generare un URL predefinito, richiamate l'URL da un browser o da un client API (ad esempio, Postman) utilizzando il seguente formato: `https://${s3-bucket-prefix}.${domain}/presign/objects/${uploaded-filename}` Sostituisci i valori segnaposto in `${s3-bucket-prefix}` e `${domain}` con i valori impostati nei passaggi precedenti.	Proprietario dell'app
Controlla il risultato.	Il risultato previsto è che dovresti ricevere un codice di stato di reindirizzamento 301 (Spostato permanentemente). Questa risposta conterrà l'URL predefinito, che dovrebbe avviare automaticamente il download del file di test.	Tecnico di test

Pulisci con Terraform

Operazione	Description	Competenze richieste

Distruggi le risorse API Gateway, Amazon S3 e Lambda.	Per eliminare le risorse, usa i seguenti comandi:
	`cd ./2.apigw_s3_lambda`
	`terraform init`
	`terraform plan --destroy --var-file=apg.tfvars`
	`terraform destroy --var-file=apg.tfvars`
		Amministratore AWS, amministratore cloud

Distruggi le risorse di rete.	Per eliminare le risorse di rete, utilizzate i seguenti comandi:
	`cd ./1.vpc_alb_ga`
	`terraform init`
	`terraform plan --destroy --var-file=apg.tfvars`
	`terraform destroy --var-file=apg.tfvars`
		Amministratore AWS, amministratore cloud

Pulisci con AWS CDK

Operazione	Description	Competenze richieste
Distruggi le pile.	Per distruggere sia il VPC che lo stack delle applicazioni, usa il seguente comando: `$ cdk destroy --all`	Amministratore AWS, amministratore cloud
Svuota ed elimina i bucket Amazon S3.	Svuota ed elimina il bucket Amazon S3 oggetto e il	Amministratore AWS, amministratore cloud

Operazione	Description	Competenze richieste

	bucket Amazon S3 di log che non vengono eliminati per impostazione predefinita.
	
	I nomi dei bucket Amazon S3 sono e. `${s3-bucket-prefix}.${domain}` `${s3-bucket-prefix}.${domain}-logs`
	
	Se preferisci usare il AWS Command Line Interface (AWS CLI) per eliminare i bucket, usa i seguenti comandi:
	
	`$ aws s3 rm s3://${s3-bucket-prefix}.${domain} --recursive`
	`$ aws s3 rb s3://${s3-bucket-prefix}.${domain} --force`
	`$ aws s3 rm s3://${s3-bucket-prefix}.${domain}-logs --recursive`
	`$ aws s3 rb s3://${s3-bucket-prefix}.${domain}-logs --force`
	
	Sostituite `${s3-bucket-prefix}` e `${domain}` con i valori impostati nei passaggi precedenti. , /p>	

Risorse correlate

AWS Blog

- Accesso a un Amazon API Gateway tramite indirizzi IP statici forniti da AWS Global Accelerator
- Genera un URL predefinito in formato modulare AWS CDK per JavaScript
- Hosting di siti Web statici HTTPS interni con ALB, S3 e PrivateLink

Crea una EventBridge connessione Amazon tra più account in un'organizzazione

Sam Wilson e Robert Stone, Amazon Web Services

Riepilogo

I sistemi distribuiti di grandi dimensioni utilizzano Amazon EventBridge per comunicare i cambiamenti di stato tra diversi account Amazon Web Services (AWS) in un' AWS Organizations organizzazione. Tuttavia, EventBridge è generalmente in grado di rivolgersi solo agli endpoint o ai consumatori della stessa Account AWS. L'eccezione è un bus di eventi in un account diverso. Quel bus di eventi è un obiettivo valido. Per utilizzare gli eventi di un bus di eventi in un altro account, gli eventi devono essere trasferiti dal bus degli eventi dell'account di origine al bus degli eventi dell'account di destinazione. Per evitare problemi nella gestione di eventi critici tra applicazioni diverse Account AWS, utilizzate l'approccio consigliato presentato in questo modello.

Questo modello illustra come implementare un'architettura basata sugli eventi EventBridge che coinvolga più persone Account AWS in un'organizzazione. AWS Organizations Il modello utilizza AWS Cloud Development Kit (AWS CDK) Toolkit e. AWS CloudFormation

EventBridge offre un bus di eventi senza server che consente di ricevere, filtrare, trasformare, indirizzare e distribuire eventi. Un componente fondamentale delle architetture basate sugli eventi, EventBridge supporta la separazione tra produttori di messaggi e consumatori di tali messaggi. In un unico account, questo è semplice. Una struttura con più account richiede considerazioni aggiuntive affinché gli eventi sull'Event Bus di un account vengano utilizzati in altri account all'interno della stessa organizzazione.

Per informazioni sulle considerazioni specifiche relative agli account per produttori e consumatori, consulta la sezione Informazioni aggiuntive.

Prerequisiti e limitazioni

Prerequisiti

- Un' AWS Organizations organizzazione con almeno due associati Account AWS
- Un ruolo AWS Identity and Access Management (IAM) in entrambi Account AWS che consente di effettuare il provisioning dell'infrastruttura in entrambi Account AWS utilizzando AWS CloudFormation

- Git installato localmente
- AWS Command Line Interface (AWS CLI) installato localmente
- AWS CDK installato localmente e avviato in entrambi Account AWS

Versioni del prodotto

Questo modello è stato creato e testato utilizzando i seguenti strumenti e versioni:

- AWS CDK Toolkit 2.126.0
- Node.js 18.19.0
- npm 10.2.3
- Python 3.12

Questo modello dovrebbe funzionare con qualsiasi versione di AWS CDK v2 o npm. Le versioni da 13.0.0 a 13.6.0 di Node.js non sono compatibili con. AWS CDK

Architecture

Architettura Target

Il diagramma seguente mostra il flusso di lavoro dell'architettura per trasferire un evento da un account e consumarlo in un altro account.

Il flusso di lavoro prevede i seguenti passaggi:

1. La AWS Lambda funzione Producer nell'account Source inserisce un evento nel bus degli EventBridge eventi dell'account.
2. La EventBridge regola cross-account indirizza l'evento a un bus di EventBridge eventi nell'account di destinazione.
3. Il bus EventBridge eventi nell'account Destination ha una regola Lambda di destinazione che richiama la funzione Consumer Lambda.

Una procedura ottimale consiste nell'utilizzare una Dead Letter Queue (DLQ) per gestire le chiamate non riuscite della funzione Consumer Lambda. Tuttavia, il DLQ è stato omesso da questa soluzione per motivi di chiarezza. Per ulteriori informazioni su come implementare un DLQ nei flussi di lavoro

e migliorare la capacità dei flussi di lavoro di ripristino in caso di errori, consulta il post sul blog Implementing error handling patterns. AWS Lambda

Automazione e scalabilità

AWS CDK esegue automaticamente il provisioning dell'architettura richiesta. EventBridge può scalare fino a migliaia di record al secondo a seconda di Regione AWS. Per ulteriori informazioni, consulta la documentazione sulle EventBridge quote Amazon.

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che ti aiuta a definire e fornire Cloud AWS l'infrastruttura in codice. Questo modello utilizza il AWS CDK Toolkit, un kit di sviluppo cloud a riga di comando che consente di interagire con AWS CDK l'app.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altro modo. Account AWS
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS Organizationsè un servizio di gestione degli account che ti aiuta a consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente.

Altri strumenti

- Node.js è un ambiente di JavaScript runtime basato sugli eventi progettato per la creazione di applicazioni di rete scalabili.
- npm è un registro software che viene eseguito in un ambiente Node.js e viene utilizzato per condividere o prendere in prestito pacchetti e gestire la distribuzione di pacchetti privati.
- Python è un linguaggio di programmazione per computer generico.

Archivio di codice

Il codice per questo modello è disponibile nel repository GitHub cross-account-eventbridge-in-organization.

Best practice

Per le migliori pratiche di utilizzo EventBridge, consulta le seguenti risorse:

- Le migliori pratiche per i modelli di EventBridge eventi di Amazon
- Le migliori pratiche per la definizione delle regole in Amazon EventBridge

Epiche

Prepara il tuo ambiente di AWS CDK distribuzione locale

Operazione	Description	Competenze richieste
Configura le credenziali locali per l'account di origine e l'account di destinazione.	Consulta la sezione Configurazione della nuova configurazione e delle nuove credenziali e utilizza il metodo di autenticazione e credenziali più adatto al tuo ambiente. **Important** Assicurati di configurare l'autenticazione sia AWS CLI per l'account di origine che per l'account di destinazione. Queste istruzioni presuppongono che tu abbia configurato due profili AWS localmente: `sourceAccount` `edestinationAccount` .	Sviluppatore di app
Bootstrap entrambi Account AWS.	Per avviare gli account, esegui i seguenti comandi:	Sviluppatore di app

Operazione	Description	Competenze richieste
	`cdk bootstrap --profile sourceAccount` `cdk bootstrap --profile destinationAccount`	

Clona il codice del pattern.	Per clonare il repository, esegui il seguente comando:
	`git clone git@github.com:aws-samples/aws-cdk-examples.git`
	Quindi, cambia la directory nella cartella del progetto appena clonata:
	`cd aws-cdk-examples/python/cross-account-eventbridge-in-organization`
		Sviluppatore di app

Esegui la distribuzione sull'account ProducerStack di origine

Operazione	Description	Competenze richieste
Modifica inserendo `cdk.json` i tuoi dati AWS Organizations e quelli del tuo account.	Nella cartella principale del progetto, apporta le seguenti modifiche a `cdk.json`: • `organization_id` – L'ID Organizations degli account coinvolti nella distribuzione • `event_bus_name` – `CrossAccount`,	Sviluppatore di app

Operazione	Description	Competenze richieste
	oppure è possibile rinominarlo • `rules[].targets[].arn` – L' Account AWS ID dell'account di consumo (account di destinazione)	
Distribuisci le ProducerStack risorse.	Esegui il comando seguente dalla directory principale del progetto: `cdk deploy ProducerStack --profile sourceAccount` Quando richiesto, accetta i nuovi ruoli IAM e le altre autorizzazioni relative alla sicurezza create tramite. AWS CloudFormation	Sviluppatore di app

Operazione	Description	Competenze richieste
Verifica che le risorse siano distribuite. ProducerStack	Per verificare le risorse, procedi come segue: 1. Nell'account Console di gestione AWS per l'account Source, scegli CloudFormation. 2. Dall'elenco delle pile, scegli ProducerStack. 3. Nella scheda Informazioni sullo stack, verifica che lo stato dello stack sia. `CREATE_COMPLETE` Facoltativamente, nella scheda Risorse, esamina le risorse configurate.	Sviluppatore di app

Esegui la distribuzione sull' ConsumerStack account di destinazione

Operazione	Description	Competenze richieste
Distribuisci le ConsumerStack risorse.	Esegui il comando seguente dalla directory principale del progetto: `cdk deploy ConsumerStack --profile destinationAccount` Quando richiesto, accetta i nuovi ruoli IAM e le altre autorizzazioni relative alla sicurezza create tramite. CloudFormation	Sviluppatore di app

Operazione	Description	Competenze richieste
Verifica che le risorse siano distribuite ConsumerStack	1. Sulla console dell'account di destinazione, scegli CloudFormation. 2. Dall'elenco degli stack, scegli ConsumerStack. 3. Nella scheda Informazioni sullo stack, verifica che lo stato dello stack sia. `CREATE_COMPLETE` Facoltativamente, nella scheda Risorse, esamina le risorse configurate.	Sviluppatore di app

Produci e consuma eventi

Operazione	Description	Competenze richieste
Invoca la funzione Producer Lambda.	1. Sulla console per l'account Source, scegli Lambda. 2. Dall'elenco delle funzioni, scegli ProducerStack-ProducerLambda XXXX (XXXX rappresenta una sequenza di caratteri generata automaticamente da AWS CDK). 3. Seleziona la scheda Test. 4. Nella sezione Test event, scegli Test. Il contenuto dell'area di testo Event JSON può essere qualsiasi JSON	Sviluppatore di app

Operazione	Description	Competenze richieste
	valido fornito alla funzione Lambda come payload. In questo caso, il codice JSON fornito di default è sufficien te. 5. Verificate che il messaggio Executing function: successed appaia in un banner verde sopra la sezione Test event.	

Operazione	Description	Competenze richieste
Verifica che l'evento sia stato ricevuto.	1. Sulla console per l'account di destinazione, scegli Lambda. 2. Dall'elenco delle funzioni, scegli ConsumerStack-ConsumerLambda XXXX (XXXX rappresenta una sequenza di caratteri generata automaticamente da AWS CDK). 3. Selezionare la scheda Monitor (Monitora). 4. Nella sezione Monitor, scegli Visualizza registri. CloudWatch 5. Nella scheda appena aperta, scegli il nome del flusso di log del flusso di log più recente. 6. Verifica che venga visualizzata un'istruzione di registro come la seguente: `[DEBUG] 2024-04-08T19:08:10.091Z 9c16844a-f9de-444d-b621-86afe64d4cc8 Event: {'version':'0', 'id':'0b9faa96-973f-8be2-ecf8-75e4f328b980', 'detail-type':'TestType', 'source':'Producer`	Sviluppatore di app

Operazione	Description	Competenze richieste
	', 'account' :'XXXXXXXXXXXX', 'time':'2024-04-08 T19:08:09Z', 'region':'us-east-1', 'resource s': [], 'detail': {'key1':'value1', 'key2':'value2', 'key3':'value3'}}	

Pulizia

Operazione	Description	Competenze richieste
Distruggi le ConsumerStack risorse.	Se utilizzi questo schema come test, pulisci le risorse distribuite per evitare di incorrere in costi aggiuntivi. Esegui il comando seguente dalla directory principale del progetto: `cdk destroy ConsumerStack --profile destinationAccount` Ti verrà richiesto di confermare l'eliminazione dello stack.	Sviluppatore di app
Distruggi le risorse. ProducerStack	Esegui il seguente comando dalla directory principale del progetto:	Sviluppatore di app

Operazione	Description	Competenze richieste
	`cdk destroy ProducerStack --profile sourceAccount` Ti verrà richiesto di confermare l'eliminazione dello stack.	

risoluzione dei problemi

Problema	Soluzione
Non è stato ricevuto alcun evento nell'account Destination.	1. Verifica che l'ID Organizations fornito sia corretto. 2. Verifica che l'account di origine faccia parte dell'organizzazione fornita. 3. Verifica che le regole del bus degli eventi nell'account di origine corrispondano alle informazioni corrette nell'account di destinazione.
L'invocazione di una funzione Lambda dalla console restituisce il seguente errore: `User: arn:aws:iam::123456789012:user/XXXXX is not authorized to perform: lambda:Invoke`	Contatta Account AWS l'amministratore per ricevere le autorizzazioni di `lambda:Invoke` azione appropriate sulla funzione `ProducerStack-ProducerLambdaXXXX` Lambda.

Risorse correlate

Riferimenti

- AWS Organizations Guida per l'utente
- Modelli di EventBridge eventi Amazon
- Regole in Amazon EventBridge

Tutorial e video

- Tutorial: creazione e configurazione di un'organizzazione
- AWS re:Invent 2023 - Modelli avanzati basati sugli eventi con Amazon (01-R) EventBridge COM3

Informazioni aggiuntive

Regola del produttore

Nell'account Source, viene creato un bus di EventBridge eventi per accettare messaggi dai produttori (come mostrato nella sezione Architettura). Su questo bus di eventi viene creata una regola con autorizzazioni IAM di accompagnamento. Le regole hanno come target il bus degli EventBridge eventi nell'account di destinazione in base alla seguente struttura: `cdk.json`

```
"rules": [
  {
    "id": "CrossAccount",
    "sources": ["Producer"],
    "detail_types": ["TestType"],
    "targets": [
      {
        "id": "ConsumerEventBus",
        "arn": "arn:aws:events:us-east-2:012345678901:event-bus/CrossAccount"
      }
    ]
  }
]
```

Per ogni bus di eventi che consuma, devono essere inclusi il pattern di eventi e il bus di eventi di destinazione.

Modello di eventi

I pattern di eventi filtrano gli eventi a cui verrà applicata questa regola. Ai fini di questo esempio, le sorgenti degli eventi e il record `detail_types` identificano gli eventi da trasmettere dal bus degli eventi dell'account di origine al bus degli eventi dell'account di destinazione.

Bus degli eventi di destinazione

Questa regola si rivolge a un bus di eventi che esiste in un altro account. Il valore completo `arn` (Amazon Resource Name) è necessario per identificare in modo univoco il bus degli eventi di

destinazione ed `id` è l'ID logico utilizzato da AWS CloudFormation. Non è necessario che il bus degli eventi di destinazione esista effettivamente al momento della creazione della regola di destinazione.

Considerazioni specifiche sull'account di destinazione

Nell'account di destinazione, viene creato un bus di EventBridge eventi per ricevere messaggi dal bus eventi dell'account di origine. Per consentire la pubblicazione degli eventi dall'account Source, devi creare una politica basata sulle risorse:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "AllowOrgToPutEvents",
    "Effect": "Allow",
    "Principal": "*",
    "Action": "events:PutEvents",
    "Resource": "arn:aws:events:us-east-2:012345678901:event-bus/CrossAccount",
    "Condition": {
      "StringEquals": {
        "aws:PrincipalOrgID": "o-XXXXXXXXX"
      }
    }
  }]
}
```

È particolarmente importante concedere l'`events:PutEvents`autorizzazione, che consente a qualsiasi altro account della stessa organizzazione di pubblicare eventi su questo bus di eventi. L'impostazione `aws:PrincipalOrgId` come ID dell'organizzazione concede le autorizzazioni necessarie.

Modello di evento

Puoi modificare il pattern di eventi incluso per adattarlo al tuo caso d'uso:

```
rule = events.Rule(
    self,
    self.id + 'Rule' + rule_definition['id'],
    event_bus=event_bus,
    event_pattern=events.EventPattern(
        source=rule_definition['sources'],
        detail_type=rule_definition['detail_types'],
    )
```

```
)
```

Per ridurre l'elaborazione non necessaria, il modello di eventi deve specificare che solo gli eventi che devono essere elaborati dall'account di destinazione vengono trasmessi al bus eventi dell'account di destinazione.

Politica basata sulle risorse

Questo esempio utilizza l'ID dell'organizzazione per controllare quali account sono autorizzati a inserire eventi sul bus eventi dell'account di destinazione. Prendi in considerazione l'utilizzo di una politica più restrittiva, come specificare l'account di origine.

EventBridge quote

Tieni presente le seguenti quote:

- La quota predefinita è 300 regole per bus di eventi. Questa può essere ampliata se necessario, ma dovrebbe adattarsi alla maggior parte dei casi d'uso.
- Il numero massimo consentito di cinque obiettivi per regola. Consigliamo agli architetti delle applicazioni di utilizzare una regola distinta per ogni account di destinazione per supportare un controllo granulare sul modello di eventi.

Distribuisci i record DynamoDB ad Amazon S3 utilizzando Kinesis Data Streams e Firehose con AWS CDK

Shashank Shrivastava e Daniel Matuki da Cunha, Amazon Web Services

Riepilogo

Questo modello fornisce codice di esempio e un'applicazione per la distribuzione di record da Amazon DynamoDB ad Amazon Simple Storage Service (Amazon S3) utilizzando Amazon Kinesis Data Streams e Amazon Data Firehose. L'approccio del pattern utilizza costrutti AWS Cloud Development Kit (AWS CDK) L3 e include un esempio di come eseguire la trasformazione dei dati AWS Lambda prima che i dati vengano consegnati al bucket S3 di destinazione sul cloud Amazon Web Services ().AWS

Kinesis Data Streams registra le modifiche a livello di elemento nelle tabelle DynamoDB e le replica nel flusso di dati Kinesis richiesto. Le applicazioni possono accedere a Kinesis Data Streams e visualizzare le modifiche a livello di elemento in tempo quasi reale. Kinesis Data Streams fornisce anche l'accesso ad altri servizi Amazon Kinesis, come Firehose e Amazon Managed Service for Apache Flink. Ciò significa che puoi creare applicazioni che forniscono dashboard in tempo reale, generare avvisi, implementare prezzi e pubblicità dinamici ed eseguire analisi sofisticate dei dati.

Puoi utilizzare questo modello per i tuoi casi d'uso di integrazione dei dati. Ad esempio, i veicoli di trasporto o le apparecchiature industriali possono inviare elevati volumi di dati a una tabella DynamoDB. Questi dati possono quindi essere trasformati e archiviati in un data lake ospitato in Amazon S3. Puoi quindi interrogare ed elaborare i dati e prevedere eventuali difetti potenziali utilizzando servizi serverless come Amazon Athena, Amazon Redshift Spectrum, Amazon Rekognition e. AWS Glue

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- AWS Command Line Interface (AWS CLI), installato e configurato. Per ulteriori informazioni, consulta la sezione Guida introduttiva AWS CLI a the nella AWS CLI documentazione.
- Node.js (18.x+) e npm, installati e configurati. Per ulteriori informazioni, consulta Download e installazione di Node.js e npm nella documentazione. `npm`

- aws-cdk (2.x+), installato e configurato. Per ulteriori informazioni, consulta la sezione Guida introduttiva a the nella documentazione. AWS CDK AWS CDK
- Il repository GitHub aws-dynamodb-kinesisfirehose-sa 3 ingestioni, clonato e configurato sul computer locale.
- Dati di esempio esistenti per la tabella DynamoDB. I dati devono utilizzare il seguente formato: `{"SourceDataId": {"S": "123"},"MessageData":{"S": "Hello World"}}`

Architecture

Il diagramma seguente mostra un esempio di flusso di lavoro per la distribuzione di record da DynamoDB ad Amazon S3 utilizzando Kinesis Data Streams e Firehose.

Il diagramma mostra il flusso di lavoro seguente:

1. I dati vengono acquisiti utilizzando Amazon API Gateway come proxy per DynamoDB. Puoi anche utilizzare qualsiasi altra fonte per importare dati in DynamoDB.
2. Le modifiche a livello di articolo vengono generate quasi in tempo reale in Kinesis Data Streams per la distribuzione ad Amazon S3.
3. Kinesis Data Streams invia i record a Firehose per la trasformazione e la distribuzione.
4. Una funzione Lambda converte i record da un formato di record DynamoDB al formato JSON, che contiene solo i nomi e i valori degli attributi degli elementi del record.

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che ti aiuta a definire e fornire l'infrastruttura cloud AWS in codice.
- AWS CDK Toolkit è un kit di sviluppo cloud a riga di comando che ti aiuta a interagire con la tua AWS CDK app.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando.
- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita tra e. Account AWS Regioni AWS

Archivio di codice

Il codice per questo pattern è disponibile nel repository a GitHub aws-dynamodb-kinesisfirehose-s3 ingestioni.

Epiche

Imposta e configura il codice di esempio

Operazione	Description	Competenze richieste

Installa le dipendenze.	Sul computer locale, installa le dipendenze dai `package.json` file contenuti nelle `sample-application` directory `pattern/aws-dynamodb-kinesisstreams-s3` and eseguendo i seguenti comandi:
	`cd <project_root>/pattern/aws-dynamodb-kinesisstreams-s3`
	`npm install && npm run build`
	`cd <project_root>/sample-application/`
	`npm install && npm run build`
		Sviluppatore di app, General AWS

Operazione	Description	Competenze richieste
Genera il CloudFormation modello.	1. Esegui il comando `cd <project_root>/sample-application/ .` 2. Esegui il `cdk synth` comando per generare il CloudFormation modello. 3. L'`AwsDynamodbKinesisfirehoseS3IngestionStack.template.json` output viene memorizzato nella `cdk.out` directory. 4. Usa AWS CDK o Console di gestione AWS in cui elaborare il modello CloudFormation.	Sviluppatore di app, General AWS, AWS DevOps

Distribuisci le risorse

Operazione	Description	Competenze richieste
Controlla e distribuisci le risorse.	1. Esegui il `cdk diff` comando per identificare i tipi di risorse creati dal AWS CDK costrutto. 2. Esegui il `cdk deploy` comando per distribuire le risorse.	Sviluppatore di app, General AWS, AWS DevOps

Inserisci dati nella tabella DynamoDB per testare la soluzione

Operazione	Description	Competenze richieste

Inserisci i tuoi dati di esempio nella tabella DynamoDB.	Invia una richiesta alla tua tabella DynamoDB eseguendo il seguente comando in: AWS CLI
	
	`aws dynamodb put-item --table-name <your_table_name> --item '{"<table_partition_key>": {"S": "<partition_key_ID>"},"MessageData":{"S": "<data>"}}'`
	
	esempio:
	
	`aws dynamodb put-item --table-name SourceData_table --item '{"SourceDataId": {"S": "123"},"MessageData":{"S": "Hello World"}}'`
	
	Per impostazione predefinita, `put-item` non restituisce alcun valore come output se l'operazione ha esito positivo. Se l'operazione non riesce, restituisce un errore. I dati vengono archiviati in DynamoDB e quindi inviati
		Sviluppatore di app

Operazione	Description	Competenze richieste
	a Kinesis Data Streams e Firehose. **Note** Utilizzate approcci diversi per aggiunger e dati in una tabella DynamoDB. Per ulteriori informazioni, consulta Caricare i dati nelle tabelle nella documentazione di DynamoDB.	
Verifica che venga creato un nuovo oggetto nel bucket S3.	Accedi Console di gestione AWS e monitora il bucket S3 per verificare che sia stato creato un nuovo oggetto con i dati che hai inviato. Per ulteriori informazi oni, GetObjectconsulta la documentazione di Amazon S3.	Sviluppatore di app, General AWS

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Eliminare le risorse.	Esegui il `cdk destroy` comando per eliminare tutte le risorse utilizzate da questo modello.	Sviluppatore di app, General AWS

Risorse correlate

- s-3 static-site-stack .ts (repository) GitHub
- aws-apigateway-dynamodb modulo (repository) GitHub
- aws-kinesisstreams-kinesisfirehose-smodulo 3 (GitHub repository)
- Modifica l'acquisizione dei dati per DynamoDB Streams (documentazione DynamoDB)
- Utilizzo di Kinesis Data Streams per acquisire le modifiche a DynamoDB (documentazione DynamoDB)

Implementa il controllo delle versioni delle API basato sul percorso utilizzando domini personalizzati in Amazon API Gateway

Corey Schnedl, Marcelo Barbosa, Mario López Martínez, Anbazhagan Ponnuswamy, Gaurav Samudra e Abhilash Vinod, Amazon Web Services

Riepilogo

Questo modello dimostra come utilizzare la funzionalità di mappatura delle API dei domini personalizzati per implementare una soluzione di controllo delle versioni delle API basata sul percorso per Amazon API Gateway.

Amazon API Gateway è un servizio completamente gestito che puoi utilizzare per creare, pubblicare, mantenere, monitorare e proteggere APIs su qualsiasi scala. Utilizzando la funzionalità di dominio personalizzato del servizio, puoi creare nomi di dominio personalizzati più semplici e intuitivi URLs da fornire agli utenti delle API. Puoi utilizzare le mappature delle API per connettere le fasi dell'API a un nome di dominio personalizzato. Dopo aver creato un nome di dominio e configurato i record DNS, utilizzi le mappature delle API per inviare traffico al tuo utente APIs tramite il tuo nome di dominio personalizzato.

Dopo che un'API diventa disponibile pubblicamente, i consumatori la utilizzano. Con l'evoluzione di un'API pubblica, anche il suo contratto di servizio si evolve per riflettere nuove funzionalità e capacità. Tuttavia, non è consigliabile modificare o rimuovere le funzionalità esistenti. Eventuali modifiche sostanziali potrebbero influire sulle applicazioni del consumatore e interromperle in fase di esecuzione. Il controllo delle versioni delle API è importante per evitare l'interruzione della compatibilità con le versioni precedenti e l'interruzione del contratto.

È necessaria una strategia chiara per il controllo delle versioni delle API per aiutare i consumatori ad adottarle. Il controllo delle versioni APIs tramite path-based URLs è l'approccio più semplice e comunemente usato. In questo tipo di controllo delle versioni, le versioni sono definite esplicitamente come parte dell'API. URIs L'esempio seguente URLs mostra come un consumatore può utilizzare l'URI per specificare una versione dell'API per la propria richiesta:

```
https://api.example.com/api/v1/orders

https://api.example.com/api/v2/orders

https://api.example.com/api/vX/orders
```

Questo modello utilizza AWS Cloud Development Kit (AWS CDK) per creare, distribuire e testare un'implementazione di esempio di una soluzione di versioning scalabile basata su percorsi per l'API. AWS CDK è un framework di sviluppo software open source per modellare e fornire le risorse delle applicazioni cloud utilizzando linguaggi di programmazione familiari.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS.
- La proprietà di un dominio è necessaria per utilizzare l'archivio di esempio di questo pattern e per utilizzare la funzionalità di dominio personalizzato di Amazon API Gateway. Puoi usare Amazon Route 53 per creare e gestire i domini per la tua organizzazione. Per informazioni su come registrare o trasferire un dominio con Route 53, consulta Registrazione di nuovi domini nella documentazione di Route 53.
- Prima di configurare un nome di dominio personalizzato per un'API, è necessario disporre di un certificato SSL/TLS. AWS Certificate Manager
- È necessario creare o aggiornare il record di risorse del provider DNS per eseguire la mappatura all'endpoint API. Senza tale mappatura, le richieste API associate al nome di dominio personalizzato non possono raggiungere API Gateway.

Limitazioni

- Un nome di dominio personalizzato deve essere unico all'interno di Regione AWS tutti Account AWS.
- Per configurare le mappature API con più livelli, è necessario utilizzare un nome di dominio personalizzato regionale e la policy di sicurezza TLS 1.2.
- In una mappatura API, il nome di dominio personalizzato e il mapping APIs devono coincidere. Account AWS
- Le mappature API devono contenere solo lettere, numeri e i seguenti caratteri: `$-_.+!*'()/`
- La lunghezza massima per il percorso in una mappatura API è di 300 caratteri.
- È possibile disporre di 200 mappature API con più livelli per ogni nome di dominio.
- È possibile mappare HTTP solo APIs a un nome di dominio personalizzato regionale con la politica di sicurezza TLS 1.2.
- Non è possibile eseguire il WebSocket APIs mapping allo stesso nome di dominio personalizzato di un'API HTTP o di un'API REST.

- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità regionale, vedi AWS Servizi per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- Questa implementazione di esempio utilizza la AWS CDK TypeScript versione 2.149.0.

Architecture

Il diagramma seguente mostra il flusso di lavoro dell'architettura.

Il diagramma illustra quanto segue:

1. L'utente API invia una richiesta ad Amazon API Gateway con un nome di dominio personalizzato.
2. API Gateway indirizza dinamicamente la richiesta dell'utente verso un'istanza e una fase appropriate di API Gateway, in base al percorso indicato nell'URL della richiesta. La tabella seguente mostra un esempio di come i diversi percorsi basati su URL possono essere instradati verso fasi specifiche per diverse istanze di API Gateway.

"Hello, World!"	Stage	Path	Endpoint predefinito
Calcolo APIv1	api	apiv1	Abilitato
Calcolo APIv2	api	apiv2	Abilitato
Calcolo X APIv	api	API Vx	Abilitato

3. L'istanza API Gateway di destinazione elabora la richiesta e restituisce il risultato all'utente.

Automazione e scalabilità

Ti consigliamo di utilizzare AWS CloudFormation stack separati per ogni versione della tua API. Con questo approccio, puoi avere un isolamento completo tra i backend a APIs cui può essere indirizzato la funzionalità di mappatura dell'API di dominio personalizzata. Un vantaggio di questo approccio è che diverse versioni dell'API possono essere distribuite o rimosse indipendentemente senza comportare il rischio di modificare un'altra API. Questo approccio aumenta la resilienza attraverso

l'isolamento degli stack. CloudFormation Inoltre, offre diverse opzioni di back-end per l'API AWS Lambda AWS Fargate, come endpoint HTTP e azioni di. Servizi AWS

Puoi utilizzare strategie di ramificazione Git, come Gitflow, in combinazione con CloudFormation stack isolati per gestire il codice sorgente distribuito in diverse versioni dell'API. Utilizzando questa opzione, puoi mantenere diverse versioni della tua API senza la necessità di duplicare il codice sorgente per le nuove versioni. Con Gitflow, puoi aggiungere tag ai commit all'interno del tuo repository git man mano che vengono eseguiti i rilasci. Di conseguenza, hai un'istantanea completa del codice sorgente relativo a una versione specifica. Poiché è necessario eseguire gli aggiornamenti, è possibile estrarre il codice di una versione specifica, apportare aggiornamenti e quindi distribuire il codice sorgente aggiornato CloudFormation nello stack che si allinea alla versione principale corrispondente. Questo approccio riduce il rischio di violare un'altra versione dell'API perché ogni versione dell'API ha un codice sorgente isolato e viene distribuita in stack separati. CloudFormation

Tools (Strumenti)

Servizi AWS

- Amazon API Gateway ti aiuta a creare, pubblicare, gestire, monitorare e proteggere REST, HTTP e WebSocket APIs su qualsiasi scala.
- AWS Certificate Manager (ACM) ti aiuta a creare, archiviare e rinnovare certificati e chiavi SSL/TLS X.509 pubblici e privati che proteggono i tuoi siti Web e le tue AWS applicazioni.
- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software open source per definire l'infrastruttura cloud in codice e fornirla tramite. CloudFormationL'implementazione di esempio di questo pattern utilizza in.AWS CDK TypeScript L'utilizzo di AWS CDK in TypeScript utilizza strumenti familiari, tra cui il TypeScript compilatore Microsoft (`tsc`), Node.js e il gestore di pacchetti del nodo (`npm`). Se preferisci, puoi usare Yarn anche se gli esempi di questo modello lo usano. `npm` I moduli che compongono la AWS Construct Library sono distribuiti tramite il `npm` repository npmjs.org.
- CloudFormationti aiuta a configurare AWS le risorse, a fornirle in modo rapido e coerente e a gestirle durante tutto il loro ciclo di vita in e. Account AWS Regioni AWS
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Route 53 è un servizio Web DNS altamente scalabile e disponibile.
- AWS WAFè un firewall per applicazioni Web che consente di monitorare le richieste HTTP e HTTPS inoltrate alle risorse protette delle applicazioni Web.

Altri strumenti

- Bruno è un client di test API open source e compatibile con git.
- cdk-nag è un'utilità open source che verifica AWS CDK le migliori pratiche delle applicazioni utilizzando pacchetti di regole.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub path-based-versioning-with-api-gateway.

Best practice

- Utilizza una solida pipeline di integrazione e distribuzione continua (CI/CD) per automatizzare il test e l'implementazione degli stack creati con. CloudFormation AWS CDK Per ulteriori informazioni relative a questa raccomandazione, consulta la AWS DevOps Well-Architected Guidance.
- AWS WAF è un firewall gestito che si integra facilmente con servizi come Amazon API Gateway. Sebbene AWS WAF non sia un componente necessario per il funzionamento di questo modello di versione, come best practice di sicurezza consigliamo di includerlo AWS WAF con API Gateway.
- Incoraggia gli utenti delle API a eseguire regolarmente l'aggiornamento alla versione più recente dell'API in modo che le versioni precedenti dell'API possano essere obsolete e rimosse in modo efficiente.
- Prima di utilizzare questo approccio in un ambiente di produzione, implementate un firewall e una strategia di autorizzazione per la vostra API.
- Implementate l'accesso alla gestione delle AWS risorse della vostra azienda Account AWS utilizzando il modello di accesso con privilegi minimi.
- Per applicare le migliori pratiche e i consigli di sicurezza per le applicazioni create con AWS CDK, si consiglia di utilizzare l'utilità cdk-nag.

Epiche

Prepara il tuo ambiente locale

Operazione	Description	Competenze richieste
Clonare il repository.	Per clonare il repository di applicazioni di esempio, esegui il seguente comando: `git clone https://github.com/aws-samples/path-based-versioning-with-api-gateway`	Sviluppatore di app
Accedere al repository clonato.	Per accedere alla posizione della cartella del repository clonata, esegui il seguente comando: `cd api-gateway-custom-domain-versioning`	Sviluppatore di app
Installare le dipendenze richieste.	Per installare le dipendenze richieste, esegui il comando seguente: `npm install`	Sviluppatore di app

Implementa lo stack di routing CloudFormation

Operazione	Description	Competenze richieste
Avvia l'implementazione dello stack di routing.	Per avviare la distribuzione dello stack di CloudFormation routing`CustomDom`	Sviluppatore di app

Operazione	Description	Competenze richieste
	`ainRouterStack` , esegui il comando seguente, sostituendolo `example.com` con il nome del dominio di tua proprietà: `npx cdk deploy CustomDomainRouterStack --parameters PrerequisiteDomainName=example.com` **Note** La distribuzione dello stack non avrà esito positivo finché la seguente attività di convalida DNS del dominio non verrà eseguita correttamente.	

Verifica della proprietà del dominio

Operazione	Description	Competenze richieste
Verifica la proprietà del tuo dominio.	Il certificato rimarrà in stato di convalida in sospeso fino a quando non dimostrerai la proprietà del dominio associato. Per dimostrare la proprietà, aggiungi i record CNAME	Sviluppatore di app, amministratore di sistema AWS, amministratore di rete

Operazione	Description	Competenze richieste
	alla zona ospitata associata al dominio. Per ulteriori informazioni, consulta la convalida DNS nella documentazione. AWS Certificate Manager L'aggiunta dei record appropriati consente la corretta `CustomDomainRouterStack` implementazione.	

Operazione	Description	Competenze richieste
Crea un record di alias che punti al tuo dominio personalizzato API Gateway.	Dopo che il certificato è stato emesso e convalidato con successo, crea un record DNS che punti all'URL del tuo dominio personalizzato Amazon API Gateway. L'URL del dominio personalizzato viene generato in modo univoco dal provisioning del dominio personalizzato e viene specificato come parametro di output. CloudFormation Di seguito è riportato un esempio del record: Politica di routing: routing semplice Nome del record: `demo.api-gateway-custom-domain-versioning.example.com` Alias: Yes (Sì) Tipo di record: un record DNS di tipo «A» che punta a una risorsa AWS Value (Valore): `d-xxxxxxxxxx.execute-api.xx-xxxx-x.amazonaws.com` TTL (secondi): 300	Sviluppatore di app, amministratore di sistema AWS, amministratore di rete

Distribuisci gli CloudFormation stack e richiama l'API

Operazione	Description	Competenze richieste

Distribuisci lo `ApiStackV1` stack.	Per distribuire lo `ApiStackV1` stack, usa il seguente comando:
	`npm run deploy-v1`
	Il seguente codice CDK aggiunge la mappatura delle API:
	`var apiMapping = new CfnApiMapping(this, "ApiMapping", { apiId: this.lambdaRestApi.restApiId, domainName: props.customDomainName.domainName, stage: "api", apiMappingKey: "api/v1", });`
		Sviluppatore di app

Implementa lo stack. `ApiStackV2`	Per distribuire lo `ApiStackV2` stack, usa il seguente comando: `npm run deploy-v2`	Sviluppatore di app
Invoca l'API.	Per richiamare l'API e testare gli endpoint dell'API utilizzando Bruno, consulta le istruzioni in Informazioni aggiuntive.	Sviluppatore di app

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Eliminare le risorse.	Per distruggere le risorse associate a questa applicazione di esempio, utilizzate il seguente comando: `npx cdk destroy --all` **Note** Assicurati di pulire tutti i record DNS di Route 53 che sono stati aggiunti manualmente per il processo di verifica della proprietà del dominio.	Sviluppatore di app

Risoluzione dei problemi

Problema	Soluzione
La distribuzione è `CustomDomainRouterStack` scaduta perché il certificato non può essere convalidato.	Assicurati di aver aggiunto i record CNAME di convalida DNS appropriati, come descritto nell'attività precedente. Il nuovo certificato potrebbe continuare a mostrare lo stato In attesa di convalida per un massimo di 30 minuti dopo l'aggiunta dei record di convalida DNS. Per ulteriori informazioni, consulta la sezione Convalida DNS nella documentazione. AWS Certificate Manager

Risorse correlate

- Implementazione del controllo delle versioni di API Gateway basato su header con Amazon CloudFront: questo post di AWS Compute Blog offre una strategia di versioning basata sugli header come alternativa alla strategia di versioning basata sul percorso delineata in questo modello.
- AWS CDK Workshop: questo workshop introduttivo si concentra sulla creazione e la distribuzione di applicazioni utilizzando il. AWS AWS Cloud Development Kit (AWS CDK) Questo workshop supporta Go, Python e. TypeScript

Informazioni aggiuntive

Testare la tua API con Bruno

Ti consigliamo di utilizzare Bruno, uno strumento di test delle API open source, per verificare che il routing basato sui percorsi funzioni correttamente per l'applicazione di esempio. Questo modello fornisce una raccolta di esempi per facilitare il test dell'API di esempio.

Per richiamare e testare la tua API, utilizza i seguenti passaggi:

1. Installa Bruno.
2. Apri Bruno.
3. Nel repository di codice di questo pattern, seleziona Bruno/Sample-API- e apri la raccolta. Gateway-Custom-Domain-Versioning
4. Per visualizzare il menu a discesa Ambienti in alto a destra dell'interfaccia utente (UI), seleziona una richiesta nella raccolta.
5. Nel menu a discesa Ambienti, seleziona Configura.
6. Sostituisci il `REPLACE_ME_WITH_YOUR_DOMAIN` valore con il tuo dominio personalizzato.
7. Scegli Salva, quindi chiudi la sezione Configurazione.
8. Per Sandbox Environment, verificate che l'opzione Attivo sia selezionata.
9. Richiama l'API utilizzando il pulsante -> per la richiesta selezionata.
10. Prendi nota di come viene gestita la convalida (passaggio di valori non numerici) in V1 rispetto a V2.

Per vedere schermate di esempio di invocazione dell'API e confronto delle convalide V1 e V2, vedi Testare l'API di esempio nel file dell'archivio di codice di questo pattern. `README.md`

Importa la libreria psycopg2 per interagire con il tuo AWS Lambda database PostgreSQL

Louis Hourcade, Amazon Web Services

Riepilogo

Psycopg è un adattatore di database PostgressSQL per Python. Gli sviluppatori utilizzano la `psycopg2` libreria per scrivere applicazioni Python che interagiscono con i database PostgreSQL.

Su Amazon Web Services (AWS), gli sviluppatori lo utilizzano anche AWS Lambdaper eseguire codice per applicazioni o servizi di backend. Lambda è un servizio di elaborazione senza server e basato sugli eventi che esegue codice senza la necessità di fornire o gestire server.

Per impostazione predefinita, quando crei una nuova funzione che utilizza un runtime Python supportato da Lambda, l'ambiente di runtime Lambda viene creato da un'immagine di base per Lambda fornita da. AWS Le librerie, come `pandas` or`psycopg2`, non sono incluse nell'immagine di base. Per utilizzare una libreria, devi raggrupparla in un pacchetto personalizzato e collegarla a Lambda.

Esistono diversi modi per raggruppare e collegare una libreria, inclusi i seguenti:

- Implementa la tua funzione Lambda da un archivio di file.zip.
- Implementa la tua funzione Lambda da un'immagine di contenitore personalizzata.
- Crea un layer Lambda e collegalo alla tua funzione Lambda.

Questo modello illustra le prime due opzioni.

Con un pacchetto di distribuzione.zip, aggiungere la `pandas` libreria alla funzione Lambda è relativamente semplice. Crea una cartella sulla tua macchina Linux, aggiungi lo script Lambda insieme alla `pandas` libreria e alle dipendenze della libreria alla cartella, comprimi la cartella e forniscila come sorgente per la tua funzione Lambda.

Sebbene l'utilizzo di un pacchetto di distribuzione.zip sia una pratica comune, questo approccio non funziona per la libreria. `psycopg2` Questo modello mostra innanzitutto l'errore che si ottiene se si utilizza un pacchetto di distribuzione.zip per aggiungere la `psycopg2` libreria alla funzione Lambda. Il modello mostra quindi come distribuire Lambda da un Dockerfile e modificare l'immagine Lambda per far funzionare la libreria. `psycopg2`

Per informazioni sulle tre risorse distribuite dal pattern, consultate la sezione Informazioni aggiuntive.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS con autorizzazioni sufficienti per distribuire le AWS risorse utilizzate da questo modello
- AWS Cloud Development Kit (AWS CDK) installato a livello globale eseguendo `npm install -g aws-cdk`
- Un client Git
- Python
- Docker

Limitazioni

- Alcune Servizi AWS non sono disponibili in tutte Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- Versione runtime di Python supportata da Lambda
- Psycopg2 versione 2.9.3
- Pandas versione 1.5.2

Architecture

Panoramica della soluzione

Per illustrare le sfide che potresti dover affrontare quando utilizzi la `psycopg2` libreria in Lambda, il pattern implementa due funzioni Lambda:

- Una funzione Lambda con il runtime Python creata da un file.zip. Le `pandas` librerie `psycopg2` and vengono installate in questo pacchetto di distribuzione.zip utilizzando pip.
- Una funzione Lambda con il runtime Python creato da un Dockerfile. Il Dockerfile installa le `pandas` librerie `psycopg2` and nell'immagine del contenitore Lambda.

La prima funzione Lambda installa la `pandas` libreria e le sue dipendenze in un file.zip e Lambda può utilizzare tale libreria.

La seconda funzione Lambda dimostra che, creando un'immagine contenitore per la funzione Lambda, è possibile eseguire le librerie and `pandas` in Lambda. `psycopg2`

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che ti aiuta a definire e fornire l'infrastruttura cloud AWS in codice.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.

Altri strumenti

- Docker è un insieme di prodotti Platform as a Service (PaaS) che utilizzano la virtualizzazione a livello di sistema operativo per fornire software in container.
- pandas è uno strumento open source basato su Python per l'analisi e la manipolazione dei dati.
- Psycopg è un adattatore di database PostgreSQL per il linguaggio Python progettato per applicazioni multithread. Questo modello utilizza Psycopg 2.
- Python è un linguaggio di programmazione per computer generico.

Deposito di codice

Il codice per questo pattern è disponibile nel repository import-psycopg2 - - su. in-lambda-to-interact with-postgres-database GitHub

Best practice

Questo modello fornisce un esempio funzionante di utilizzo AWS CDK per creare una funzione Lambda da un Dockerfile. Se riutilizzi questo codice nella tua applicazione, assicurati che le risorse distribuite soddisfino tutti i requisiti di sicurezza. Utilizza strumenti come Checkov, che analizza le configurazioni dell'infrastruttura cloud per individuare eventuali errori di configurazione prima che l'infrastruttura venga implementata.

Epiche

Clona il repository e configura la distribuzione

Operazione	Description	Competenze richieste

Clonare il repository.	Per clonare il GitHub repository sul computer locale, esegui i seguenti comandi:
	`git clone https://github.com/aws-samples/import-psycopg2-in-lambda-to-interact-with-postgres-database.git`
	`cd AWS-lambda-psycopg2`
		Informazioni generali su AWS

Configura la tua distribuzione.	Modifica il `app.py` file con informazioni su Account AWS:
	`aws_acccount = "AWS_ACCOUNT_ID"`
	`region = "AWS_REGION"`
	`# Select the CPU architecture you are using to build the image (ARM or X86)`
	`architecture = "ARM"`
		Informazioni generali su AWS

Avvia il tuo account AWS e distribuisci l'applicazione

Operazione	Description	Competenze richieste
Bootstrap il tuo. Account AWS	Se non hai ancora avviato il tuo ambiente AWS, esegui i seguenti comandi con le AWS	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	credenziali del tuo account: AWS `cdk bootstrap aws://<tooling-account-id>/<aws-region>`	
Implementa il codice.	Per distribuire l' AWS CDK applicazione, esegui il comando seguente: `cdk deploy AWSLambdaPyscopg2`	Informazioni generali su AWS

Testa le funzioni Lambda dalla Console di gestione AWS

Operazione	Description	Competenze richieste
Prova la funzione Lambda creata dal file.zip.	Per testare la funzione Lambda creata dal file.zip, procedi come segue: 1. Accedi alla console e apri la console Lambda all'indirizzo. https://console.aws.amazon.com/lambda/ 2. Seleziona la funzione `lambda-from-zip` Lambda. 3. Crea un evento di test per richiamare la funzione. 4. Quando viene richiamata, la funzione dovrebbe generare	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	un errore che include il seguente messaggio: `"errorMessage": Unable to import module 'lambda_code': libpq.so.5: cannot open shared object, "stackTrace": [] "errorType": Runtime.ImportModuleError",` 5. Apri la CloudWatch console Amazon all'indirizzo https://console.aws.amazon.com/cloudwatch/. I CloudWatch registri mostrano che la `pandas` libreria è stata importata correttamente ma che l'importazione della `psycopg2` libreria non è riuscita. Poiché Lambda non trova le librerie PostgreSQL richieste nell'immagine predefinita, non può utilizzare la libreria. `psycopg2`	

Operazione	Description	Competenze richieste

Prova la funzione Lambda creata dal Dockerfile.	Per utilizzare la `psycopg2` libreria all'interno della tua funzione Lambda, devi modificare l'Amazon Machine Image (AMI) Lambda.
	
	Per testare la funzione Lambda creata dal Dockerfile, procedi come segue:
	
	1. Accedi alla console e apri la console Lambda.
	2. Seleziona la funzione `lambda-from-docker` Lambda.
	3. Crea un evento di test per richiamare la funzione.
	4. Quando viene richiamata, la funzione dovrebbe essere eseguita correttamente.
	
	Il codice seguente mostra il Dockerfile creato dal AWS CDK modello:
	
	`# Start from lambda Python3.13 image`
	`FROM public.ecr.aws/lambda/python:3.13`
	
	`# Copy the lambda code, together with its requirements`
	`COPY lambda/requirements.txt ${LAMBDA_TASK_ROOT}`
		Informazioni generali su AWS

Operazione	Description	Competenze richieste

	`COPY lambda/lambda_code .py ${LAMBDA_TASK_ROOT }`
	
	`# Install postgresql- devel in your image RUN yum install -y gcc postgresql-devel`
	
	`# install the requireme nts for the Lambda code RUN pip3 install -r requirements.txt --target "${LAMBDA _TASK_ROOT}"`
	
	`# Command can be overwritten by providing a different command in the template directly. CMD ["lambda_code.hand ler"]`
	
	Il Dockerfile prende l'immagin e AWS Lambda fornita per il runtime Python e installa postgresql-devel, che contiene le librerie necessarie per compilare applicazioni che interagiscono direttamente con il server di gestione PostgreSQL. Il Dockerfile installa anche le librerie and, indicate nel file. `pandas psycopg2 requireme nts.txt`	

Risorse correlate

- AWS CDK documentazione
- AWS Lambda documentazione

Informazioni aggiuntive

In questo modello, il AWS CDK modello fornisce uno AWS stack con tre risorse:

- Un ruolo AWS Identity and Access Management (IAM) per le funzioni Lambda.
- Una funzione Lambda con un runtime Python. La funzione viene distribuita dal pacchetto di distribuzione. `Constructs/lambda/lambda_deploy.zip`
- Una funzione Lambda con un runtime Python. La funzione viene distribuita dal Dockerfile nella cartella `Constructs`

Lo script per entrambe le funzioni Lambda verifica se le `psycopg2` librerie `pandas` and sono state importate correttamente:

```
import pandas
print("pandas successfully imported")

import psycopg2
print("psycopg2 successfully imported")

def handler(event, context):
    """Function that checks whether psycopg2  and pandas are successfully imported or
 not"""
    return {"Status": "psycopg2 and pandas successfully imported"}
```

Il pacchetto `lambda_deploy.zip` di distribuzione è creato con lo script `Constructs/lambda/build.sh` bash. Questo script crea una cartella, copia lo script Lambda, installa le `psycopg2` librerie `pandas` and e genera il file.zip. Per generare tu stesso il file.zip, esegui questo script bash e ridistribuisci lo stack. AWS CDK

Il Dockerfile inizia con l'immagine di base AWS fornita per Lambda con un runtime Python. Il Dockerfile installa le `psycopg2` librerie `pandas` and sopra l'immagine predefinita.

Integra Amazon API Gateway con Amazon SQS per gestire REST asincrono APIs

Natalia Colantonio Favero e Gustavo Martim, Amazon Web Services

Riepilogo

Quando si distribuisce REST APIs, a volte è necessario esporre una coda di messaggi che le applicazioni client possono pubblicare. Ad esempio, potreste avere problemi con la latenza di terze parti APIs e i ritardi nelle risposte, oppure potreste voler evitare i tempi di risposta delle query del database o evitare il ridimensionamento del server quando il numero di richieste simultanee è elevato. APIs In questi scenari, le applicazioni client che pubblicano nella coda devono solo sapere che l'API ha ricevuto i dati, non cosa accade dopo la ricezione dei dati.

Questo modello crea un endpoint API REST utilizzando Amazon API Gateway per inviare un messaggio ad Amazon Simple Queue Service (Amazon SQS). Crea un' easy-to-implementintegrazione tra i due servizi che evita l'accesso diretto alla coda SQS.

Prerequisiti e limitazioni

- Un account attivo AWS

Architecture

Il diagramma illustra questi passaggi:

1. Richiedi un endpoint dell'API POST REST utilizzando uno strumento come Postman, un'altra API o altre tecnologie.
2. API Gateway pubblica un messaggio, che viene ricevuto nel corpo della richiesta, sulla coda.
3. Amazon SQS riceve il messaggio e invia una risposta ad API Gateway con un codice di successo o di errore.

Tools (Strumenti)

- Amazon API Gateway ti aiuta a creare, pubblicare, gestire, monitorare e proteggere REST, HTTP e WebSocket APIs su qualsiasi scala.

- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon Simple Queue Service (Amazon SQS) fornisce una coda ospitata sicura, durevole e disponibile che ti aiuta a integrare e disaccoppiare sistemi e componenti software distribuiti.

Epiche

Crea una coda SQS

Operazione	Description	Competenze richieste
Crea una coda.	Per creare una coda SQS che riceva i messaggi dall'API REST: 1. Accedi al tuo Account AWS. 2. Apri la console Amazon SQS all'indirizzo. https://console.aws.amazon.com/sqs/ 3. Scegliere Crea coda. 4. Nella pagina Crea coda, scegli quella corretta Regione AWS dall'elenco a discesa Region. 5. Per Tipo, mantieni l'impostazione predefinita (Standard). 6. Inserisci un Nome per la coda. 7. Mantieni i valori predefiniti per tutte le altre impostazioni. 8. Scegliere Crea coda.	Sviluppatore di app

Fornisci l'accesso ad Amazon SQS

Operazione	Description	Competenze richieste
Crea un ruolo IAM.	Questo ruolo IAM offre alle risorse API Gateway l'accesso completo ad Amazon SQS. 1. Aprire la console IAM all'indirizzo https://console.aws.amazon.com/iam/. 2. Nel pannello di navigazione seleziona Ruoli, quindi Crea ruolo. 3. Per Tipo di entità attendibile, seleziona Servizio AWS. 4. Per Use Case, scegli API Gateway dall'elenco a discesa, quindi scegli Avanti, Avanti. 5. Per Nome ruolo, inserisci AWSGatewayRoleForSQS e una descrizione opzionale, quindi scegli Crea ruolo. 6. Nel riquadro Ruoli, cerca AWSGatewayRoleForSQS e seleziona la relativa casella di controllo. 7. Nella sezione Policy di autorizzazioni, seleziona Aggiungi autorizzazioni, Collega policy. 8. Cerca Amazon SQSFullAccess e selezionalo.	Sviluppatore di app, amministratore AWS

Operazione	Description	Competenze richieste
	9. Scegli Add Permissions (Aggiungi autorizzazioni). 10. Nella sezione Riepilogo di AWSGatewayRoleForSQS, copia l'Amazon Resource Number (ARN). Utilizzer ai questo ID in una fase successiva.	

Crea un'API REST

Operazione	Description	Competenze richieste
Crea un'API REST.	Questa è l'API REST a cui vengono inviate le richieste HTTP. 1. Apri la console API Gateway all'indirizzo https://console.aws.amazon.com/apigateway/. 2. Nella sezione API REST, scegli Build. 3. Per il nome dell'API, inserisci un nome e una descrizione opzionale per l'API, mantieni tutte le altre impostazioni predefinite, quindi scegli Crea API.	Sviluppatore di app
Connetti API Gateway ad Amazon SQS.	Questo passaggio consente al messaggio di fluire dall'interno del corpo della richiesta HTTP ad Amazon SQS.	Sviluppatore di app

Operazione	Description	Competenze richieste
	1. Nella console API Gateway, scegli l'API che hai creato. 2. Nella pagina Risorse, nella sezione Metodi, scegli Crea metodo. 3. Per Metodo HTTP scegliere POST. 4. Per Tipo di integrazione, scegliete Servizio AWS. 5. Per Regione AWS, scegli la regione in cui hai creato la coda SQS. 6. Per Servizio AWS, scegli Simple Queue Service (SQS). 7. Per il metodo HTTP, scegli POST. 8. Per il tipo di azione, scegli Usa l'override del percorso. 9. <name of SQS queue>Per Path override, inserisci /<AWS account ID>. 10. Per il ruolo Execution, incolla l'ARN del ruolo creato in precedenza. 11. Scegli Crea metodo.	

Prova l'API REST

Operazione	Description	Competenze richieste

Prova l'API REST.	Esegui un test per verificare la configurazione mancante:
	1. Nella console API Gateway, scegli l'API REST che hai creato.
	2. Nel riquadro Risorse, scegli il metodo POST.
	3. Seleziona la scheda Test. (Usa la freccia destra se la scheda non è visualizzata.)
	4. Per Request body, incolla il seguente codice JSON:
	`{ "message": "lorem ipsum" }`
	5. Scegli Test (Esegui test).
	Riceverai un errore simile al seguente:
	`<UnknownOperationException/>`
		Sviluppatore di app

Modifica l'integrazione dell'API per inoltrare correttamente la richiesta ad Amazon SQS.	Completa la configurazione per correggere l'errore di integrazione: 1. Nella console API Gateway, scegli l'API che hai creato, quindi scegli POST.	Sviluppatore di app

Operazione	Description	Competenze richieste
	2. La sezione Method Execution mostra la mappatura visiva tra API Gateway e Amazon SQS. Da questa sezione, scegli Richiesta di integrazione, quindi scegli Modifica. 3. Espandi la sezione delle intestazioni HTTP, quindi scegli il parametro Aggiungi intestazione della richiesta. • Per Nome, specificate Content-Type. • Per Mapped from, inserisci 'application/ '. x-www-form-urlencoded Assicurati di includere le virgolette singole. • Seleziona la casella di controllo Caching. 4. Espandi la sezione Modelli di mappatura. • Scegliere Add mapping template (Aggiungi modello di mappatura). • Per Tipo di contenuto, inserisci application/json. • Per Template body, incolla questo codice: `Action=SendMessage&MessageBody=$input.body`	

Operazione	Description	Competenze richieste
	• Scegli Save (Salva).	

Operazione	Description	Competenze richieste
Testa e convalida il messaggio in Amazon SQS.	Esegui un test per confermare che il test è stato completato con successo: 1. Nella console API Gateway, scegli l'API REST che hai creato. 2. Nel riquadro Risorse, scegli il metodo POST. 3. Seleziona la scheda Test. (Usa la freccia destra se la scheda non è visualizzata.) 4. Per Request body, incolla il seguente codice JSON: `{ "message": "lorem ipsum" }` 5. Scegli Test (Esegui test). 6. Apri la console Amazon SQS. 7. Nel riquadro di navigazione, scegli Code, quindi scegli la tua coda. 8. Scegli Invia e ricevi messaggi. 9. Scegli Sondaggio per i messaggi. 10. Scegliere Message (Messaggio). Dovrebbe mostrare quanto segue:	Sviluppatore di app

Operazione	Description	Competenze richieste
	`Body { "message": "lorem ipsum" }`	

Operazione	Description	Competenze richieste

Prova API Gateway con un carattere speciale.	Esegui un test che includa caratteri speciali (come &) che non sono accettabili in un messaggio:
	1. Nella console API Gateway, scegli la tua API.
	2. Ripeti il test del passaggio precedente utilizzando il seguente codice JSON:
	`{ "message": "lorem ipsum &" }`
	3. Scegli Test (Esegui test).
	Riceverai un errore come il seguente:
	`{ "Error": { "Code": "AccessDenied", "Message": "Access to the resource https://sqs.us-east-2.amazonaws.com/976166761794/Apg2 is denied.", "Type": "Sender" }, "RequestId": "e83c9c67-bcf6-5e9a-91e9-c737094b17ab"`
		Sviluppatore di app

Operazione	Description	Competenze richieste
	<pre>}</pre> Questo perché i caratteri speciali non sono supportati per impostazione predefinita nel corpo del messaggio. Nel passaggio successivo, configurerai API Gateway per supportare i caratteri speciali. Per ulteriori informazioni sulle conversioni dei tipi di contenuto, consulta la documentazione di API Gateway.	

Operazione	Description	Competenze richieste

Modifica la configurazione dell'API per supportare i caratteri speciali.	Modifica la configurazione per accettare caratteri speciali nel messaggio:
	1. Nella console API Gateway, scegli l'API che hai creato, quindi scegli POST.
	2. Scegli Richiesta di integrazione, quindi seleziona Modifica.
	3. Modifica la gestione dei contenuti in Converti in testo.
	4. Nella sezione Modelli di mappatura:
	• Per Tipo di contenuto, inserisci application/json.
	• Per il corpo del modello, specificare:
	`Action=SendMessage&MessageBody=$util.urlEncode($input.body)`
	• Scegli Save (Salva).
	5. Seleziona la scheda Test.
	6. Per Request body, inserisci il codice JSON di cui hai parlato in precedenza:
	`{ " message": "lorem ipsum &" }`
		Sviluppatore di app

Operazione	Description	Competenze richieste
	7. Scegli Test (Esegui test). 8. Apri la console Amazon SQS. 9. Seleziona la coda, quindi scegli Invia e ricevi messaggi, Sondaggio per i messaggi, Invia messaggio come prima. Il nuovo messaggio deve includere il carattere speciale.	

Implementa l'API REST

Operazione	Description	Competenze richieste
Implementa l'API.	Per distribuire l'API REST: 1. Apri la console API Gateway. 2. Scegliere l'API. 3. Seleziona Deploy API (Distribuisci API). Per ulteriori informazioni su questo passaggio, consulta la documentazione di API Gateway.	Sviluppatore di app
Esegui il test con uno strumento esterno.	Esegui un test con uno strumento esterno per confermare che il messaggio sia stato ricevuto correttam ente:	Sviluppatore di app

Operazione	Description	Competenze richieste
	1. Apri uno strumento come Postman, Insomnia o cURL. 2. Esegui la tua API. 3. Apri la console Amazon SQS. 4. Seleziona la tua coda. 5. Carica i messaggi per vedere il nuovo messaggio.	

Pulizia

Operazione	Description	Competenze richieste
Eliminare l'API.	Nella console API Gateway, scegli l'API che hai creato, quindi scegli Elimina.	Sviluppatore di app
Elimina il ruolo IAM.	Sulla console IAM, nel riquadro Ruoli, seleziona AWSGatewayRoleForSQS, quindi scegli Elimina.	Sviluppatore di app
Elimina la coda SQS.	Sulla console Amazon SQS, nel riquadro Code, scegli la coda SQS che hai creato, quindi scegli Elimina.	Sviluppatore di app

Risorse correlate

- SQS- SendMessage (documentazione API Gateway)
- Conversioni dei tipi di contenuto in API Gateway (documentazione API Gateway)
- variabili $util (documentazione API Gateway)

- Come posso integrare un'API REST di API Gateway con Amazon SQS e risolvere gli errori più comuni? (AWS Re:post articolo)

Elabora gli eventi in modo asincrono con Amazon API Gateway e AWS Lambda

Andrea Meroni, Mariem Kthiri, Nadim Majed e Michael Wallner, Amazon Web Services

Riepilogo

Amazon API Gateway è un servizio completamente gestito che gli sviluppatori possono utilizzare per creare, pubblicare, mantenere, monitorare e proteggere APIs su qualsiasi scala. Gestisce le attività legate all'accettazione e all'elaborazione di fino a centinaia di migliaia di chiamate API simultanee.

Una quota di servizio importante di API Gateway è il timeout di integrazione. Il timeout è il tempo massimo in cui un servizio di backend deve restituire una risposta prima che l'API REST restituisca un errore. Il limite rigido di 29 secondi è generalmente accettabile per i carichi di lavoro sincroni. Tuttavia, tale limite rappresenta una sfida per gli sviluppatori che desiderano utilizzare API Gateway con carichi di lavoro asincroni.

Questo modello mostra un'architettura di esempio per elaborare gli eventi in modo asincrono utilizzando API Gateway e. AWS Lambda L'architettura supporta l'esecuzione di processi di elaborazione della durata fino a 15 minuti e utilizza un'API REST di base come interfaccia.

Projen viene utilizzato per configurare l'ambiente di sviluppo locale e per distribuire l'architettura di esempio su un target Account AWS, in combinazione con AWS Cloud Development Kit (AWS CDK) Toolkit, Docker e Node.js. Projen configura automaticamente un ambiente virtuale Python con pre-commit e gli strumenti utilizzati per il controllo della qualità del codice, la scansione di sicurezza e il test delle unità. Per ulteriori informazioni, consulta la sezione Strumenti.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- I seguenti strumenti installati sulla workstation:
  - AWS Cloud Development Kit (AWS CDK) Toolkit versione 2.85.0
  - Docker versione 20.10.21
  - Node.js versione 18.13.0
  - Versione del progetto 0.71.111

- Python versione 3.9.16

Limitazioni

- La durata massima di un processo è limitata dalla durata massima delle funzioni Lambda (15 minuti).
- Il numero massimo di richieste di lavoro simultanee è limitato dalla concorrenza riservata della funzione Lambda.

Architecture

Il diagramma seguente mostra l'interazione dell'API jobs con le funzioni Lambda di elaborazione degli eventi e gestione degli errori, con gli eventi archiviati in un archivio di eventi Amazon. EventBridge

Un tipico flusso di lavoro include i seguenti passaggi:

1. Ci si autentica con AWS Identity and Access Management (IAM) e si ottengono le credenziali di sicurezza.
2. Si invia una `POST` richiesta HTTP all'endpoint dell'API `/jobs` jobs, specificando i parametri del processo nel corpo della richiesta.
3. L'API jobs, che è un'API REST di API Gateway, ti restituisce una risposta HTTP che contiene l'identificatore del lavoro.
4. L'API jobs richiama in modo asincrono la funzione Lambda di elaborazione degli eventi.
5. La funzione di elaborazione degli eventi elabora l'evento e quindi inserisce i risultati del lavoro nella tabella dei job Amazon DynamoDB.
6. Invia una `GET` richiesta HTTP all'endpoint dell'API `/jobs/{jobId}` jobs, con l'identificatore del lavoro del passaggio 3 as. `{jobId}`
7. L'API jobs interroga la tabella `jobs` DynamoDB per recuperare i risultati del lavoro.
8. L'API jobs restituisce una risposta HTTP che contiene i risultati del lavoro.
9. Se l'elaborazione dell'evento non riesce, la funzione di elaborazione degli eventi invia l'evento alla funzione di gestione degli errori.
10. La funzione di gestione degli errori inserisce i parametri del lavoro nella tabella DynamoDB`jobs`.
11. È possibile recuperare i parametri del processo inviando una `GET` richiesta HTTP all'endpoint dell'API jobs. `/jobs/{jobId}`

12. Se la gestione degli errori fallisce, la funzione di gestione degli errori invia l'evento a un archivio di eventi. EventBridge

    È possibile riprodurre gli eventi archiviati utilizzando. EventBridge

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che consente di definire e fornire l' Cloud AWS infrastruttura nel codice.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella shell della riga di comando.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, funzioni Lambda, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altri. Account AWS
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.

Altri strumenti

- autopep8 formatta automaticamente il codice Python in base alla guida di stile Python Enhancement Proposal (PEP) 8.
- Bandit scansiona il codice Python per trovare problemi di sicurezza comuni.
- Commitizen è un controllore e generatore di commit Git. `CHANGELOG`
- cfn-lint è un linter AWS CloudFormation
- Checkov è uno strumento statico di analisi del codice che verifica eventuali configurazioni errate di sicurezza e conformità dell'infrastruttura come codice (IaC).
- jq è uno strumento a riga di comando per l'analisi di JSON.
- Postman è una piattaforma API.
- pre-commit è un gestore di hook Git.

- Projen è un generatore di progetti.
- pytest è un framework Python per scrivere test piccoli e leggibili.

Archivio di codice

Questo codice di architettura di esempio è disponibile nell'archivio GitHub Asynchronous Event Processing with API Gateway e Lambda.

Best practice

- Questa architettura di esempio non include il monitoraggio dell'infrastruttura distribuita. Se il tuo caso d'uso richiede il monitoraggio, valuta la possibilità di aggiungere CDK Monitoring Constructs o un'altra soluzione di monitoraggio.
- Questa architettura di esempio utilizza le autorizzazioni IAM per controllare l'accesso all'API jobs. Chiunque sia autorizzato a presumere che `JobsAPIInvokeRole` sarà in grado di richiamare l'API jobs. Pertanto, il meccanismo di controllo degli accessi è binario. Se il tuo caso d'uso richiede un modello di autorizzazione più complesso, valuta l'utilizzo di un meccanismo di controllo degli accessi diverso.
- Quando un utente invia una `POST` richiesta HTTP all'endpoint dell'API `/jobs` jobs, i dati di input vengono convalidati a due livelli diversi:
  - Amazon API Gateway è responsabile della prima convalida della richiesta.
  - La funzione di elaborazione degli eventi esegue la seconda richiesta.

    Non viene eseguita alcuna convalida quando l'utente effettua una `GET` richiesta HTTP all'endpoint dell'API `/jobs/{jobId}` jobs. Se il tuo caso d'uso richiede un'ulteriore convalida dell'input e un maggiore livello di sicurezza, valuta l'utilizzo di AWS WAF per proteggere la tua API.

Epiche

Configura l'ambiente

Operazione	Description	Competenze richieste
Clonare il repository.	Per clonare il repository localmente, esegui il seguente comando:	DevOps ingegnere

Operazione	Description	Competenze richieste
	`git clone https://github.com/aws-samples/asynchronous-event-processing-api-gateway-lambda-cdk.git`	

Configura il progetto.	Cambia la directory nella radice del repository e configura l'ambiente virtuale Python e tutti gli strumenti usando Projen:
	`cd asynchronous-event-processing-api-gateway-api-gateway-lambda-cdk`
	`npx projen`
		DevOps ingegnere

Installa i ganci di pre-commit.	Per installare gli hook di pre-commit, procedi come segue:
	1. Attiva l'ambiente virtuale Python:
	`source .env/bin/activate`
	2. Installa gli hook di pre-commit:
	`pre-commit install`
	`pre-commit install --hook-type commit-msg`
		DevOps ingegnere

Implementa l'architettura di esempio

Operazione	Description	Competenze richieste
Bootstrap. AWS CDK	Per eseguire il bootstrap AWS CDK in tuo Account AWS, esegui il seguente comando: `AWS_PROFILE=$YOUR_AWS_PROFILE npx projen bootstrap`	AWS DevOps
Implementa l'architettura di esempio.	Per implementare l'architettura di esempio nella tua Account AWS, esegui il seguente comando: `AWS_PROFILE=$YOUR_AWS_PROFILE npx projen deploy`	AWS DevOps

Testa l'architettura

Operazione	Description	Competenze richieste
Installa i prerequisiti del test.	Installa sulla tua workstation the AWS Command Line Interface (AWS CLI), Postman e jq. L'uso di Postman per testare questa architettura di esempio è consigliato ma non obbligatorio. Se scegli uno strumento di test delle API alternativo, assicurati che supporti l'autenticazione AWS Signature	DevOps ingegnere

Operazione	Description	Competenze richieste
	Version 4 e fai riferimento agli endpoint API esposti che possono essere ispezionati esportando l'API REST.	
Supponiamo il`JobsAPIInvokeRole` .	Supponiamo `JobsAPIInvokeRole` che sia stato stampato come output del comando deploy: `CREDENTIALS=$(AWS_PROFILE=$<YOUR_AWS_PROFILE> aws sts assume-role \ --no-cli-pager \ --role-arn $<JOBS_API_INVOKE_ROLE_ARN> \ --role-session-name JobsAPIInvoke) export AWS_ACCESS_KEY_ID=$(cat $CREDENTIALS \| jq '.Credentials''.AccessKeyId') export AWS_SECRET_ACCESS_KEY=$(cat $CREDENTIALS \| jq '.Credentials''.SecretAccessKey') export AWS_SESSION_TOKEN==$(cat $CREDENTIALS | jq '.Credentials''.SessionToken')`	AWS DevOps

Operazione	Description	Competenze richieste
Configura Postman.	1. Per importare la collezione Postman inclusa nel repository, segui le istruzioni nella documentazione di Postman. 2. Imposta le `JobsAPI` variabili con i seguenti valori: • `accessKey` – Il valore dell'`Credentials.AccessKeyId` attributo del `assume-role` comando • `baseUrl`– Il valore dell'`JobsApiJobsAPIEndpoint` output del comando deploy, senza la barra finale • `region`– Il valore del Regione AWS luogo in cui è stata implementata l'architettura di esempio • `seconds`– Il valore del parametro di input per il lavoro di esempio. Deve essere un numero intero positivo • `secretKey` – Il valore dell'`Credentials.SecretAccessKey` attributo	AWS DevOps

Operazione	Description	Competenze richieste
	del comando `assume-role` • `sessionToken` – Il valore dell'`Credentials.SessionToken` attributo del `assume-role` comando	
Prova l'architettura di esempio.	Per testare l'architettura di esempio, invia le richieste all'API jobs. Per ulteriori informazioni, consulta la documentazione di Postman.	DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
La distruzione e la successiva ridistribuzione dell'architettura di esempio non riescono perché il gruppo di log di Amazon CloudWatch Logs esiste `/aws/apigateway/JobsAPIAccessLogs` già.	1. Se necessario, esporta i dati di registro in Amazon S3. 2. Elimina il gruppo di CloudWatch log Logs. `/aws/apigateway/JobsAPIAccessLogs` 3. Ridistribuite l'architettura di esempio.

Risorse correlate

- Modello di mappatura API Gateway e riferimento alla variabile di registrazione degli accessi
- Imposta la chiamata asincrona della funzione Lambda di backend

Elabora gli eventi in modo asincrono con Amazon API Gateway e Amazon DynamoDB Streams

Andrea Meroni, Mariem Kthiri, Nadim Majed, Alessandro Trisolini e Michael Wallner, Amazon Web Services

Riepilogo

Amazon API Gateway è un servizio completamente gestito che gli sviluppatori possono utilizzare per creare, pubblicare, mantenere, monitorare e proteggere APIs su qualsiasi scala. Gestisce le attività legate all'accettazione e all'elaborazione di fino a centinaia di migliaia di chiamate API simultanee.

Una quota di servizio importante di API Gateway è il timeout di integrazione. Il timeout è il tempo massimo in cui un servizio di backend deve restituire una risposta prima che l'API REST restituisca un errore. Il limite rigido di 29 secondi è generalmente accettabile per i carichi di lavoro sincroni. Tuttavia, tale limite rappresenta una sfida per gli sviluppatori che desiderano utilizzare API Gateway con carichi di lavoro asincroni.

Questo modello mostra un'architettura di esempio per l'elaborazione asincrona degli eventi utilizzando API Gateway, Amazon DynamoDB Streams e. AWS Lambda L'architettura supporta l'esecuzione di processi di elaborazione parallela con gli stessi parametri di input e utilizza un'API REST di base come interfaccia. In questo esempio, l'utilizzo di Lambda come backend limita la durata dei processi a 15 minuti. È possibile evitare questo limite utilizzando un servizio alternativo per elaborare gli eventi in arrivo (ad esempio,). AWS Fargate

Projen viene utilizzato per configurare l'ambiente di sviluppo locale e per distribuire l'architettura di esempio su un target Account AWS, in combinazione con AWS Cloud Development Kit (AWS CDK) Toolkit, Docker e Node.js. Projen configura automaticamente un ambiente virtuale Python con pre-commit e gli strumenti utilizzati per il controllo della qualità del codice, la scansione di sicurezza e il test delle unità. Per ulteriori informazioni, consulta la sezione Strumenti.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- I seguenti strumenti installati sulla workstation:
  - AWS Cloud Development Kit (AWS CDK) Toolkit versione 2.85.0 o successiva

- Docker versione 20.10.21 o successiva
- Node.js versione 18 o successiva
- Versione Projen 0.71.111 o successiva
- Python versione 3.9.16 o successiva

Limitazioni

- Il numero massimo consigliato di lettori per DynamoDB Streams è due per evitare la limitazione.
- La durata massima di un processo è limitata dalla durata massima delle funzioni Lambda (15 minuti).
- Il numero massimo di richieste di lavoro simultanee è limitato dalla concorrenza riservata delle funzioni Lambda.

Architecture

Architettura

Il diagramma seguente mostra l'interazione dell'API dei job con DynamoDB Streams e le funzioni Lambda di elaborazione degli eventi e gestione degli errori, con gli eventi archiviati in un archivio di eventi Amazon. EventBridge

Un tipico flusso di lavoro include i seguenti passaggi:

1. Ci si autentica con AWS Identity and Access Management (IAM) e si ottengono le credenziali di sicurezza.
2. Si invia una `POST` richiesta HTTP all'endpoint dell'API `/jobs` jobs, specificando i parametri del processo nel corpo della richiesta.
3. L'API jobs ti restituisce una risposta HTTP che contiene l'identificatore del lavoro.
4. L'API jobs inserisce i parametri del processo nella tabella `jobs_table` Amazon DynamoDB.
5. Il flusso `jobs_table` DynamoDB della tabella DynamoDB richiama le funzioni Lambda di elaborazione degli eventi.
6. Le funzioni Lambda per l'elaborazione degli eventi elaborano l'evento e quindi inseriscono i risultati del lavoro nella tabella DynamoDB. `jobs_table` Per garantire risultati coerenti, le funzioni di elaborazione degli eventi implementano un meccanismo di blocco ottimistico.

7. Si invia una `GET` richiesta HTTP all'endpoint dell'API `/jobs/{jobId}` jobs, con l'identificatore del lavoro del passaggio 3 as. `{jobId}`
8. L'API jobs interroga la tabella `jobs_table` DynamoDB per recuperare i risultati del lavoro.
9. L'API jobs restituisce una risposta HTTP che contiene i risultati del lavoro.
10. Se l'elaborazione dell'evento non riesce, la mappatura dei sorgenti della funzione di elaborazione degli eventi invia l'evento all'argomento Amazon Simple Notification Service (Amazon SNS) sulla gestione degli errori.
11. L'argomento SNS sulla gestione degli errori invia l'evento in modo asincrono alla funzione di gestione degli errori.
12. La funzione di gestione degli errori inserisce i parametri del lavoro nella tabella DynamoDB`jobs_table`.

    È possibile recuperare i parametri del processo inviando una `GET` richiesta HTTP all'endpoint dell'API jobs. `/jobs/{jobId}`
13. Se la gestione degli errori fallisce, la funzione di gestione degli errori invia l'evento a un archivio Amazon EventBridge .

    Puoi riprodurre gli eventi archiviati utilizzando. EventBridge

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che ti aiuta a definire e fornire l'infrastruttura cloud AWS in codice.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, funzioni AWS Lambda, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altri account AWS.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.

- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.

Altri strumenti

- autopep8 formatta automaticamente il codice Python in base alla guida di stile Python Enhancement Proposal (PEP) 8.
- Bandit scansiona il codice Python per trovare problemi di sicurezza comuni.
- Commitizen è un controllore e generatore di commit Git. `CHANGELOG`
- cfn-lint è un linter AWS CloudFormation
- Checkov è uno strumento statico di analisi del codice che verifica eventuali configurazioni errate di sicurezza e conformità dell'infrastruttura come codice (IaC).
- jq è uno strumento a riga di comando per l'analisi di JSON.
- Postman è una piattaforma API.
- pre-commit è un gestore di hook Git.
- Projen è un generatore di progetti.
- pytest è un framework Python per scrivere test piccoli e leggibili.

Archivio di codice

Questo codice di architettura di esempio è disponibile nel repository GitHub Asynchronous Processing with API Gateway e DynamoDB Streams.

Best practice

- Questa architettura di esempio non include il monitoraggio dell'infrastruttura distribuita. Se il tuo caso d'uso richiede il monitoraggio, valuta la possibilità di aggiungere CDK Monitoring Constructs o un'altra soluzione di monitoraggio.
- Questa architettura di esempio utilizza le autorizzazioni IAM per controllare l'accesso all'API jobs. Chiunque sia autorizzato a presumere che `JobsAPIInvokeRole` sarà in grado di richiamare l'API jobs. Pertanto, il meccanismo di controllo degli accessi è binario. Se il tuo caso d'uso richiede un modello di autorizzazione più complesso, valuta l'utilizzo di un meccanismo di controllo degli accessi diverso.
- Quando un utente invia una `POST` richiesta HTTP all'endpoint dell'API `/jobs` jobs, i dati di input vengono convalidati a due livelli diversi:

- API Gateway è responsabile della prima convalida della richiesta.
- La funzione di elaborazione degli eventi esegue la seconda richiesta.

  Non viene eseguita alcuna convalida quando l'utente effettua una `GET` richiesta HTTP all'endpoint dell'API `/jobs/{jobId}` jobs. Se il tuo caso d'uso richiede un'ulteriore convalida degli input e un maggiore livello di sicurezza, valuta l'utilizzo AWS WAF per proteggere la tua API.

- Per evitare il throttling, la documentazione di DynamoDB Streams scoraggia gli utenti dal leggere con più di due consumatori dello stesso shard dello stesso stream. Per ridimensionare il numero di consumatori, consigliamo di utilizzare Amazon Kinesis Data Streams.
- In questo esempio è stato utilizzato il blocco ottimistico per garantire aggiornamenti coerenti degli elementi nella tabella DynamoDB`jobs_table`. A seconda del requisito del caso d'uso, potrebbe essere necessario implementare meccanismi di blocco più affidabili, come il blocco pessimistico.

Epiche

Configura l'ambiente

Operazione	Description	Competenze richieste
Clonare il repository.	Per clonare il repository localmente, esegui il seguente comando: `git clone https://github.com/aws-samples/asynchronous-event-processing-api-gateway-dynamodb-streams-cdk.git`	DevOps ingegnere
Configura il progetto.	Cambia la directory nella radice del repository e configura l'ambiente virtuale Python e tutti gli strumenti usando Projen:	DevOps ingegnere

Operazione	Description	Competenze richieste
	`cd asynchronous-event-processing-api-gateway-api-gateway-dynamodb-streams-cdk` `npx projen`	

Installa i ganci di pre-commit.	Per installare gli hook di pre-commit, procedi come segue:
	1. Attiva l'ambiente virtuale Python:
	`source .env/bin/activate`
	2. Installa gli hook di pre-commit:
	`pre-commit install`
	`pre-commit install --hook-type commit-msg`
		DevOps ingegnere

Implementa l'architettura di esempio

Operazione	Description	Competenze richieste
Bootstrap. AWS CDK	Per eseguire il bootstrap AWS CDKin tuo Account AWS, esegui il seguente comando: `AWS_PROFILE=$YOUR_AWS_PROFILE npx projen bootstrap`	AWS DevOps
Implementa l'architettura di esempio.	Per implementare l'architettura di esempio nella tua Account	AWS DevOps

Operazione	Description	Competenze richieste
	AWS, esegui il seguente comando: `AWS_PROFILE=$YOUR_ AWS_PROFILE npx projen deploy`	

Testa l'architettura

Operazione	Description	Competenze richieste
Installa i prerequisiti del test.	Installa sulla tua workstation the AWS Command Line Interface (AWS CLI), Postman e jq. L'uso di Postman per testare questa architettura di esempio è consigliato ma non obbligatorio. Se scegli uno strumento di test delle API alternativo, assicurati che supporti l'autenticazione AWS Signature versione 4 e fai riferimento agli endpoint API esposti che possono essere ispezionati esportando l'API REST.	DevOps ingegnere
Supponiamo il `JobsAPIInvokeRole` .	Supponiamo `JobsAPIInvokeRole` che sia stato stampato come output del `deploy` comando: `CREDENTIALS=$(AWS_ PROFILE=$<YOUR_AWS`	AWS DevOps

Operazione	Description	Competenze richieste
	_PROFILE> aws sts assume-role \ --no-cli-pager \ --role-arn $<JOBS_API_INVOKE_ROLE_ARN> \ --role-session-name JobsAPIInvoke) export AWS_ACCESS_KEY_ID=$(cat $CREDENTIALS \| jq '.Credentials''.AccessKeyId') export AWS_SECRET_ACCESS_KEY=$(cat $CREDENTIALS \| jq '.Credentials''.SecretAccessKey') export AWS_SESSION_TOKEN==$(cat $CREDENTIALS | jq '.Credentials''.SessionToken')	

Operazione	Description	Competenze richieste

Configura Postman.	• Per importare la collezione Postman inclusa nel repository, segui le istruzioni nella documentazione di Postman.
	• Imposta le `JobsAPI` variabili con i seguenti valori:
	  • `accessKey` – Il valore dell'`Credentials.AccessKeyId` attributo del `assume-role` comando.
	  • `baseUrl`– Il valore dell'`JobsApiJobsAPIEndpoint` output del `deploy` comando, senza la barra finale.
	  • `region`– Il valore del Regione AWS luogo in cui è stata implementata l'architettura di esempio.
	  • `seconds`– Il valore del parametro di input per il lavoro di esempio. Deve essere un numero intero positivo.
	  • `secretKey` – Il valore dell'`Credentials.SecretAccessKey` attributo
		AWS DevOps

Operazione	Description	Competenze richieste
	del `assume-role` comando. • `sessionToken` – Il valore dell'`Credentials.SessionToken` attributo del `assume-role` comando.	
Prova l'architettura di esempio.	Per testare l'architettura di esempio, invia le richieste all'API jobs. Per ulteriori informazioni, consulta la documentazione di Postman.	DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
La distruzione e la successiva ridistribuzione dell'architettura di esempio non riescono perché il gruppo di log di Amazon CloudWatch Logs esiste `/aws/apigateway/JobsAPIAccessLogs` già.	1. Se necessario, esporta i dati di log in Amazon Simple Storage Service (Amazon S3). 2. Elimina il gruppo di CloudWatch log Logs. `/aws/apigateway/JobsAPIAccessLogs` 3. Ridistribuite l'architettura di esempio.

Risorse correlate

- Modello di mappatura API Gateway e riferimento alla variabile di registrazione degli accessi
- Modifica l'acquisizione dei dati per DynamoDB Streams
- Blocco ottimistico con numero di versione
- Utilizzo di Kinesis Data Streams per acquisire le modifiche a DynamoDB

Elabora gli eventi in modo asincrono con Amazon API Gateway, Amazon SQS e AWS Fargate

Andrea Meroni, Mariem Kthiri, Nadim Majed, Alessandro Trisolini e Michael Wallner, Amazon Web Services

Riepilogo

Amazon API Gateway è un servizio completamente gestito che gli sviluppatori possono utilizzare per creare, pubblicare, mantenere, monitorare e proteggere APIs su qualsiasi scala. Gestisce le attività necessarie per accettare ed elaborare fino a centinaia di migliaia di chiamate API simultanee.

Una quota di servizio importante di API Gateway è il timeout di integrazione. Il timeout è il tempo massimo in cui un servizio di backend deve restituire una risposta prima che l'API REST restituisca un errore. Il limite rigido di 29 secondi è generalmente accettabile per i carichi di lavoro sincroni. Tuttavia, tale limite rappresenta una sfida per gli sviluppatori che desiderano utilizzare API Gateway con carichi di lavoro asincroni.

Questo modello mostra un'architettura di esempio per elaborare gli eventi in modo asincrono utilizzando API Gateway, Amazon Simple Queue Service (Amazon SQS) e. AWS Fargate L'architettura supporta l'esecuzione di processi di elaborazione senza restrizioni di durata e utilizza un'API REST di base come interfaccia.

Projen viene utilizzato per configurare l'ambiente di sviluppo locale e per distribuire l'architettura di esempio su una destinazione Account AWS, in combinazione con Docker e Node.js. AWS Cloud Development Kit (AWS CDK) Projen configura automaticamente un ambiente virtuale Python con pre-commit e gli strumenti utilizzati per il controllo della qualità del codice, la scansione di sicurezza e il test delle unità. Per ulteriori informazioni, consulta la sezione Strumenti.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- I seguenti strumenti installati sulla workstation:
  - AWS Cloud Development Kit (AWS CDK) Toolkit versione 2.85.0 o successiva
  - Docker versione 20.10.21 o successiva
  - Node.js versione 18 o successiva
  - Versione Projen 0.71.111 o successiva

- Python versione 3.9.16 o successiva

Limitazioni

- I lavori simultanei sono limitati a 500 attività al minuto, che è il numero massimo di attività che Fargate può fornire.

Architecture

Il diagramma seguente mostra l'interazione dell'API jobs con la tabella `jobs` Amazon DynamoDB, il servizio Fargate di elaborazione degli eventi e la funzione di gestione degli errori. AWS Lambda Gli eventi vengono archiviati in un archivio di EventBridge eventi Amazon.

Un tipico flusso di lavoro include i seguenti passaggi:

1. Ci si autentica con AWS Identity and Access Management (IAM) e si ottengono le credenziali di sicurezza.
2. Si invia una `POST` richiesta HTTP all'endpoint dell'API `/jobs` jobs, specificando i parametri del processo nel corpo della richiesta.
3. L'API jobs, che è un'API REST di API Gateway, ti restituisce una risposta HTTP che contiene l'identificatore del lavoro.
4. L'API jobs invia un messaggio alla coda SQS.
5. Fargate estrae il messaggio dalla coda SQS, elabora l'evento e quindi inserisce i risultati del lavoro nella tabella DynamoDB. `jobs`
6. Si invia una `GET` richiesta HTTP all'endpoint dell'API `/jobs/{jobId}` jobs, con l'identificatore del lavoro del passaggio 3 as. `{jobId}`
7. L'API jobs interroga la tabella `jobs` DynamoDB per recuperare i risultati del lavoro.
8. L'API jobs restituisce una risposta HTTP che contiene i risultati del lavoro.
9. Se l'elaborazione dell'evento non riesce, la coda SQS invia l'evento alla coda di lettere morte (DLQ).
10. Un EventBridge evento avvia la funzione di gestione degli errori.
11. La funzione di gestione degli errori inserisce i parametri del lavoro nella tabella DynamoDB`jobs`.
12. È possibile recuperare i parametri del processo inviando una `GET` richiesta HTTP all'endpoint dell'API jobs. `/jobs/{jobId}`

13. Se la gestione degli errori fallisce, la funzione di gestione degli errori invia l'evento a un archivio. EventBridge

    È possibile riprodurre gli eventi archiviati utilizzando. EventBridge

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che consente di definire e fornire l' Cloud AWS infrastruttura nel codice.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- AWS Fargateti aiuta a eseguire contenitori senza dover gestire server o istanze Amazon Elastic Compute Cloud EC2 (Amazon). Viene utilizzato insieme ad Amazon Elastic Container Service (Amazon ECS).
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, funzioni Lambda, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altri. Account AWS
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Simple Queue Service (Amazon SQS) fornisce una coda ospitata sicura, durevole e disponibile che ti aiuta a integrare e disaccoppiare sistemi e componenti software distribuiti.

Altri strumenti

- autopep8 formatta automaticamente il codice Python in base alla guida di stile Python Enhancement Proposal (PEP) 8.
- Bandit scansiona il codice Python per trovare problemi di sicurezza comuni.
- Commitizen è un controllore e generatore di commit Git. `CHANGELOG`
- cfn-lint è un linter AWS CloudFormation
- Checkov è uno strumento statico di analisi del codice che verifica eventuali configurazioni errate di sicurezza e conformità dell'infrastruttura come codice (IaC).

- jq è uno strumento a riga di comando per l'analisi di JSON.
- Postman è una piattaforma API.
- pre-commit è un gestore di hook Git.
- Projen è un generatore di progetti.
- pytest è un framework Python per scrivere test piccoli e leggibili.

Archivio di codice

Questo codice di architettura di esempio è disponibile nell'archivio GitHub Asynchronous Processing with API Gateway e SQS.

Best practice

- Questa architettura di esempio non include il monitoraggio dell'infrastruttura distribuita. Se il tuo caso d'uso richiede il monitoraggio, valuta la possibilità di aggiungere CDK Monitoring Constructs o un'altra soluzione di monitoraggio.
- Questa architettura di esempio utilizza le autorizzazioni IAM per controllare l'accesso all'API jobs. Chiunque sia autorizzato a presumere che `JobsAPIInvokeRole` sarà in grado di richiamare l'API jobs. Pertanto, il meccanismo di controllo degli accessi è binario. Se il tuo caso d'uso richiede un modello di autorizzazione più complesso, valuta l'utilizzo di un meccanismo di controllo degli accessi diverso.
- Quando un utente invia una `POST` richiesta HTTP all'endpoint dell'API `/jobs` jobs, i dati di input vengono convalidati a due livelli diversi:
  - API Gateway è responsabile della prima convalida della richiesta.
  - La funzione di elaborazione degli eventi esegue la seconda richiesta.

    Non viene eseguita alcuna convalida quando l'utente effettua una `GET` richiesta HTTP all'endpoint dell'API `/jobs/{jobId}` jobs. Se il tuo caso d'uso richiede un'ulteriore convalida degli input e un maggiore livello di sicurezza, valuta l'utilizzo AWS WAF per proteggere la tua API.

Epiche

Configura l'ambiente

Operazione	Description	Competenze richieste
Clonare il repository.	Per clonare il repository localmente, esegui il seguente comando: `git clone https://github.com/aws-samples/asynchronous-event-processing-api-gateway-sqs-cdk.git`	DevOps ingegnere

Configura il progetto.	Cambia la directory nella radice del repository e configura l'ambiente virtuale Python e tutti gli strumenti usando Projen:
	`cd asynchronous-event-processing-api-gateway-api-gateway-sqs-cdk`
	`npx projen`
		DevOps ingegnere

Installa i ganci di pre-commit.	Per installare gli hook di pre-commit, procedi come segue: 1. Attiva l'ambiente virtuale Python: `source .env/bin/activate` 2. Installa gli hook di pre-commit:	DevOps ingegnere

Operazione	Description	Competenze richieste
	`pre-commit install` `pre-commit install --hook-type commit-msg`	

Implementa l'architettura di esempio

Operazione	Description	Competenze richieste
Bootstrap. AWS CDK	Per eseguire il bootstrap AWS CDKin tuo Account AWS, esegui il seguente comando: `AWS_PROFILE=$YOUR_AWS_PROFILE npx projen bootstrap`	AWS DevOps
Implementa l'architettura di esempio.	Per implementare l'architettura di esempio nella tua Account AWS, esegui il seguente comando: `AWS_PROFILE=$YOUR_AWS_PROFILE npx projen deploy`	AWS DevOps

Testa l'architettura

Operazione	Description	Competenze richieste
Installa i prerequisiti del test.	Installa sulla tua workstation the AWS Command Line Interface (AWS CLI), Postman e jq.	DevOps ingegnere

Operazione	Description	Competenze richieste
	L'uso di Postman per testare questa architettura di esempio è consigliato ma non obbligato rio. Se scegli uno strumento di test delle API alternativo, assicurati che supporti l'autenticazione AWS Signature Version 4 e fai riferimento agli endpoint API esposti che possono essere ispezionati esportando l'API REST.	

Operazione	Description	Competenze richieste
Supponiamo il`JobsAPIInvokeRole` .	Supponiamo `JobsAPIInvokeRole` che sia stato stampato come output del `deploy` comando: `CREDENTIALS=$(AWS_PROFILE=$<YOUR_AWS_PROFILE> aws sts assume-role \ --no-cli-pager \ --role-arn $<JOBS_API_INVOKE_ROLE_ARN> \ --role-session-name JobsAPIInvoke) export AWS_ACCESS_KEY_ID=$(cat $CREDENTIALS \| jq '.Credentials''.AccessKeyId') export AWS_SECRET_ACCESS_KEY=$(cat $CREDENTIALS \| jq '.Credentials''.SecretAccessKey') export AWS_SESSION_TOKEN==$(cat $CREDENTIALS | jq '.Credentials''.SessionToken')`	AWS DevOps

Operazione	Description	Competenze richieste

Configura Postman.	• Per importare la collezione Postman inclusa nel repository, segui le istruzioni nella documentazione di Postman.
	• Imposta le `JobsAPI` variabili con i seguenti valori:
	  • `accessKey` – Il valore dell'`Credentials.AccessKeyId` attributo del `assume-role` comando.
	  • `baseUrl`– Il valore dell'`JobsApiJobsAPIEndpoint` output del `deploy` comando, senza la barra finale.
	  • `region`– Il valore del Regione AWS luogo in cui è stata implementata l'architettura di esempio.
	  • `seconds`– Il valore del parametro di input per il lavoro di esempio. Deve essere un numero intero positivo.
	  • `secretKey` – Il valore dell'`Credentials.SecretAccessKey` attributo
		AWS DevOps

Operazione	Description	Competenze richieste
	del `assume-role` comando. • `sessionToken` – Il valore dell'`Credentials.SessionToken` attributo del `assume-role` comando.	
Prova l'architettura di esempio.	Per testare l'architettura di esempio, invia le richieste all'API jobs. Per ulteriori informazioni, consulta la documentazione di Postman.	DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
La distruzione e la successiva ridistribuzione dell'architettura di esempio non riescono perché il gruppo di log di Amazon CloudWatch Logs esiste `/aws/apigateway/JobsAPIAccessLogs` già.	1. Se necessario, esporta i dati di log in Amazon Simple Storage Service (Amazon S3). 2. Elimina il gruppo di CloudWatch log Logs. `/aws/apigateway/JobsAPIAccessLogs` 3. Ridistribuite l'architettura di esempio.
La distruzione e la successiva ridistribuzione dell'architettura di esempio non riescono perché il gruppo di log CloudWatch Logs esiste già. `/aws/ecs/EventProcessingServiceLogs`	1. Se necessario, esporta i dati di registro in Amazon S3. 2. Elimina il gruppo di CloudWatch log Logs `/aws/ecs/EventProcessingServiceLogs.` 3. Ridistribuite l'architettura di esempio.

Risorse correlate

- Modello di mappatura API Gateway e riferimento alla variabile di registrazione degli accessi
- Come posso integrare un'API REST di API Gateway con Amazon SQS e risolvere gli errori più comuni?

Esegui le attività di AWS Systems Manager Automation in modo sincrono da AWS Step Functions

Elie El Khoury, Amazon Web Services

Riepilogo

Questo modello spiega come eseguire l'integrazione con. AWS Step Functions AWS Systems Manager Utilizza le integrazioni dei servizi AWS SDK per chiamare l'startAutomationExecutionAPI Systems Manager con un token di attività da un flusso di lavoro di una macchina a stati e si interrompe fino a quando il token non restituisce una chiamata riuscita o non riuscita. Per dimostrare l'integrazione, questo modello implementa un wrapper di documenti di automazione (runbook) attorno al documento or e lo utilizza per chiamare `AWS-RunShellScript` o in modo `AWS-RunPowerShellScript` sincrono. `.waitForTaskToken` `AWS-RunShellScript` `AWS-RunPowerShellScript` Per ulteriori informazioni sulle integrazioni dei servizi AWS SDK in Step Functions, consulta la Guida per gli AWS Step Functions sviluppatori.

Step Functions è un servizio di flusso di lavoro visivo a basso codice che puoi utilizzare per creare applicazioni distribuite, automatizzare i processi IT e aziendali e creare pipeline di dati e apprendimento automatico utilizzando i servizi. AWS I flussi di lavoro gestiscono gli errori, i nuovi tentativi, la parallelizzazione, le integrazioni dei servizi e l'osservabilità in modo da poterti concentrare su una logica aziendale di maggior valore.

L'automazione, una funzionalità di AWS Systems Manager, semplifica le attività comuni di manutenzione, distribuzione e riparazione per Servizi AWS Amazon Elastic Compute Cloud (Amazon EC2), Amazon Relational Database Service (Amazon RDS), Amazon Redshift e Amazon Simple Storage Service (Amazon S3). L'automazione ti offre un controllo granulare sulla concomitanza delle tue automazioni. Ad esempio, è possibile specificare quante risorse indirizzare contemporaneamente e quanti errori possono verificarsi prima che un'automazione venga interrotta.

Per i dettagli di implementazione, inclusi i passaggi del runbook, i parametri e gli esempi, consulta la sezione Informazioni aggiuntive.

Prerequisiti e limitazioni

Prerequisiti

- Un account attivo AWS

- AWS Identity and Access Management Autorizzazioni (IAM) per accedere a Step Functions and Systems Manager
- Un' EC2 istanza con Systems Manager Agent (SSM Agent) installato sull'istanza
- Un profilo di istanza IAM per Systems Manager collegato all'istanza in cui intendi eseguire il runbook
- Un ruolo Step Functions con le seguenti autorizzazioni IAM (che seguono il principio del privilegio minimo):

```
{
            "Effect": "Allow",
            "Action": "ssm:StartAutomationExecution",
            "Resource": "*"
 }
```

Versioni del prodotto

- Schema del documento SSM versione 0.3 o successiva
- SSM Agent versione 2.3.672.0 o successiva

Architecture

Stack tecnologico Target

- AWS Step Functions
- Automazione di AWS Systems Manager

Architettura di destinazione

Automazione e scalabilità

- Questo modello fornisce un AWS CloudFormation modello che è possibile utilizzare per distribuire i runbook su più istanze. (Vedi l'archivio di implementazione di GitHub Step Functions and Systems Manager.)

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita in tutte le regioni Account AWS .
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Step Functionsè un servizio di orchestrazione senza server che consente di combinare AWS Lambda funzioni e altro Servizi AWS per creare applicazioni aziendali critiche.
- AWS Systems Managerconsente di gestire le applicazioni e l'infrastruttura in esecuzione in. Cloud AWS Semplifica la gestione delle applicazioni e delle risorse, riduce i tempi di rilevamento e risoluzione dei problemi operativi e aiuta a gestire le AWS risorse in modo sicuro su larga scala.

Codice

Il codice per questo modello è disponibile nell'archivio di implementazione di GitHub Step Functions and Systems Manager.

Epiche

Crea runbook

Operazione	Description	Competenze richieste
Scarica il CloudFormation modello.	Scarica il `ssm-automation-documents.cfn.json` modello dalla `cloudformation` cartella del GitHub repository.	AWS DevOps
Crea runbook.	Accedi a Console di gestione AWS, apri la CloudFormation console e distribuisci il modello. Per ulteriori informazioni sulla distribuzione dei CloudFormation modelli,	AWS DevOps

Operazione	Description	Competenze richieste
	consulta Creazione di uno stack sulla CloudFormation console nella documentazione. CloudFormation Il CloudFormation modello distribuisce tre risorse: • `SfnRunCommandByInstanceIds` — Runbook che consente di eseguire `AWS-RunShellScript` o utilizzare un'`AWS-RunPowerShellScript` istanza. IDs • `SfnRunCommandByTargets` — Runbook che consente di eseguire `AWS-RunShellScript` o `AWS-RunPowerShellScript` utilizzando obiettivi. • `SSMSyncRole` — Il ruolo IAM assunto dai runbook.	

Crea un esempio di macchina a stati

Operazione	Description	Competenze richieste
Crea una macchina a stati di test.	Segui le istruzioni contenute nella Guida per gli AWS Step Functions sviluppatori per creare ed eseguire una macchina a stati. Per la definizione, utilizzate il	AWS DevOps

Operazione	Description	Competenze richieste
	codice seguente. Assicurati di aggiornare il `InstanceIds` valore con l'ID di un'istanza valida abilitata per Systems Manager nel tuo account. `{ "Comment": "A description of my state machine", "StartAt": "StartAutomationWaitForCallBack", "States": { "StartAutomationWaitForCallBack": { "Type": "Task", "Resource": "arn:aws:states:::aws-sdk:ssm:startAutomationExecution.waitForTaskToken", "Parameters": { "DocumentName": "SfnRunCommandByInstanceIds", "Parameters": { "InstanceIds": ["i-1234567890abcdef0"], "taskToken.$": "States.Array($$.Task.Token)", "workingDirectory": ["/home/ssm-user/"], "Commands": [`	

Operazione	Description	Competenze richieste

	<pre>"echo \"This is a test running automation waitForTaskToken\" >> automation.log",
	"sleep 100"
	],
	"executionTimeout": [
	"10800"
	],
	"deliveryTimeout": [
	"30"
	],
	"shell": [
	"Shell"
	]
	}
	},
	"End": true
	}
	}
	}</pre>	

	Questo codice richiama il runbook per eseguire due comandi che dimostrano la `waitForTaskToken` chiamata a Systems Manager Automation. Il valore del `shell` parametro (`ShelloPowerShell`) determina se il documento di automazione viene eseguito `AWS-RunShellScript` o`AWS-RunPowerShellScript` .	

Operazione	Description	Competenze richieste
	L'attività scrive «Questo è un waitForTask token di automazione in esecuzione di test» nel `/home/ssm-user/automation.log` file, quindi rimane inattiva per 100 secondi prima di rispondere con il token dell'attività e rilasciare l'attività successiva del flusso di lavoro. Se invece vuoi chiamare il `SfnRunCommandByTargets` runbook, sostituisci la `Parameters` sezione del codice precedente con la seguente: `"Parameters": { "Targets": [ { "Key": "InstanceIds", "Values": [ "i-02573cafcfEXAMPLE", "i-0471e04240EXAMPLE" ] } ],`	

Operazione	Description	Competenze richieste
Aggiorna il ruolo IAM per la macchina a stati.	Il passaggio precedente crea automaticamente un ruolo IAM dedicato per la macchina a stati. Tuttavia, non concede le autorizzazioni per chiamare il runbook. Aggiorna il ruolo aggiungendo le seguenti autorizzazioni: `{ "Effect": "Allow", "Action": "ssm:StartAutomationExecution", "Resource": "*" }`	AWS DevOps
Convalida le chiamate sincrone.	Esegui la macchina a stati per convalidare la chiamata sincrona tra Step Functions e Systems Manager Automation. Per un esempio di output, vedere la sezione Informazioni aggiuntive.	AWS DevOps

Risorse correlate

- Guida introduttiva a AWS Step Functions (Guida per AWS Step Functions sviluppatori)
- Attendi una richiamata con il task token (Guida per AWS Step Functions gli sviluppatori, modelli di integrazione dei servizi)
- chiamate API send_task_success e send_task_failure (documentazione Boto3)
- AWS Systems Manager Automazione (guida AWS Systems Manager per l'utente)

Informazioni aggiuntive

Dettagli di implementazione

Questo modello fornisce un CloudFormation modello che implementa due runbook di Systems Manager:

- `SfnRunCommandByInstanceIds`esegue il `AWS-RunPowerShellScript` comando `AWS-RunShellScript` or utilizzando instance. IDs
- `SfnRunCommandByTargets`esegue il `AWS-RunPowerShellScript` comando `AWS-RunShellScript` or utilizzando target.

Ogni runbook implementa quattro passaggi per eseguire una chiamata sincrona quando si utilizza l'`.waitForTaskToken`opzione in Step Functions.

Fase	Azione	Description
1	`Branch`	Controlla il valore del `shell` parametro (`ShelloPowerShell`) per decidere se eseguire per Linux o `AWS-RunShellScript` `AWS-RunPowerShellScript` per Windows.
2	`RunCommand_Shell` o `RunCommand_PowerShell`	Accetta diversi input ed esegue il `RunPowerShellScript` comando `RunShellScript` or. Per ulteriori informazioni, consulta la scheda Dettagli per il documento `RunCommand_Shell` o `RunCommand_PowerShell` Automation nella console Systems Manager.

3	`SendTaskFailure`	Viene eseguito quando il passaggio 2 viene interrotto o annullato. Richiama l'API send_task_failure di Step Functions, che accetta tre parametri come input: il token passato dalla macchina a stati, l'errore di errore e una descrizione della causa dell'errore.
4	`SendTaskSuccess`	Viene eseguito quando il passaggio 2 ha esito positivo. Chiama l'API Step Functions send_task_success, che accetta il token passato dalla macchina a stati come input.

Parametri del runbook

`SfnRunCommandByInstanceIds`runbook:

Nome del parametro	Tipo	Facoltativo o richiesto	Description
`shell`	Stringa	Richiesto	La shell delle istanze per decidere se eseguire `AWS-RunShellScript` per Linux o `AWS-RunPowerShellScript` per Windows.
`deliveryTimeout`	Numero intero	Facoltativo	Il tempo, in secondi, di attesa per l'invio di un comando all'agente SSM su un'istanza. Questo parametro

			ha un valore minimo di 30 (0,5 minuti) e un valore massimo di 2592000 (720 ore).
`executionTimeout`	Stringa	Facoltativo	Il tempo, in secondi, necessario per il completamento di un comando prima che venga considerato non riuscito. Il valore predefinito è 3600 (1 ora). Il valore massimo è 172800 (48 ore).
`workingDirectory`	Stringa	Facoltativo	Il percorso alla directory di lavoro nell'istanza.
`Commands`	StringList	Richiesto	Lo script o il comando di shell da eseguire.
`InstanceIds`	StringList	Richiesto	Le IDs istanze in cui si desidera eseguire il comando.
`taskToken`	Stringa	Richiesto	Il token di attività da utilizzare per le risposte di callback.

`SfnRunCommandByTargets`runbook:

Name	Tipo	Facoltativo o richiesto	Description
`shell`	Stringa	Richiesto	La shell delle istanze per decidere se eseguire `AWS-RunSh`

			`ellScript` per Linux o `AWS-RunPowerShellScript` per Windows.
`deliveryTimeout`	Numero intero	Facoltativo	Il tempo, in secondi, di attesa per l'invio di un comando all'agente SSM su un'istanza. Questo parametro ha un valore minimo di 30 (0,5 minuti) e un valore massimo di 2592000 (720 ore).
`executionTimeout`	Numero intero	Facoltativo	Il tempo, in secondi, necessario per il completamento di un comando prima che venga considerato non riuscito. Il valore predefinito è 3600 (1 ora). Il valore massimo è 172800 (48 ore).
`workingDirectory`	Stringa	Facoltativo	Il percorso alla directory di lavoro nell'istanza.
`Commands`	StringList	Richiesto	Lo script o il comando di shell da eseguire.

`Targets`	MapList	Richiesto	Una matrice di criteri di ricerca che identifica le istanze utilizzando coppie chiave-valore specificate dall'utente. Ad esempio: `[{"Key":"InstanceIds","Values":["i-02573cafcfEXAMPLE","i-0471e04240EXAMPLE"]}]`
`taskToken`	Stringa	Richiesto	Il token di attività da utilizzare per le risposte di callback.

Esempio di output

La tabella seguente fornisce un esempio di output della funzione step. Mostra che il tempo di esecuzione totale è superiore a 100 secondi tra il passaggio 5 (`TaskSubmitted`) e il passaggio 6 (`TaskSucceeded`). Ciò dimostra che la funzione step ha atteso il completamento del `sleep 100` comando prima di passare all'attività successiva del flusso di lavoro.

ID	Tipo	Fase	Risorsa	Tempo trascorso (ms)	Time stamp
1	`ExecutionStarted`		-	0	11 marzo 2022 14:50:34.303
2	`TaskStateEntered`	`StartAutomationWai`	-	40	11 marzo 2022 14:50:34.343

		tForCallBack			
3	TaskScheduled	StartAutomationWaitForCallBack	-	40	11 marzo 2022 14:50:34.343
4	TaskStarted	StartAutomationWaitForCallBack	-	154	11 marzo 2022 14:50:34.457
5	TaskSubmitted	StartAutomationWaitForCallBack	-	657	11 marzo 2022 14:50:34.960
6	TaskSucceeded	StartAutomationWaitForCallBack	-	103835	11 marzo 2022 14:52:18.138
7	TaskStateExited	StartAutomationWaitForCallBack	-	103860	11 marzo 2022 02:52:18.163
8	ExecutionSucceeded		-	103897	11 marzo 2022 14:52:18.200

Esegui letture parallele di oggetti S3 usando Python in una funzione AWS Lambda

Eduardo Bortoluzzi, Amazon Web Services

Riepilogo

Puoi utilizzare questo modello per recuperare e riepilogare un elenco di documenti dai bucket Amazon Simple Storage Service (Amazon S3) in tempo reale. Il modello fornisce codice di esempio per oggetti di lettura parallela dai bucket S3 su Amazon Web Services ()AWS. Il modello mostra come eseguire in modo efficiente attività I/O associate con AWS Lambda funzioni utilizzando Python.

Una società finanziaria ha utilizzato questo modello in una soluzione interattiva per approvare o rifiutare manualmente le transazioni finanziarie correlate in tempo reale. I documenti relativi alle transazioni finanziarie sono stati archiviati in un bucket S3 relativo al mercato. Un operatore ha selezionato un elenco di documenti dal bucket S3, ha analizzato il valore totale delle transazioni calcolate dalla soluzione e ha deciso di approvare o rifiutare il batch selezionato.

Le attività legate all'I/O supportano più thread. In questo codice di esempio, concurrent.futures.ThreadPoolExecutorviene utilizzato con un massimo di 30 thread simultanei, anche se le funzioni Lambda supportano fino a 1.024 thread (uno di questi thread è il processo principale). Questo limite è dovuto al fatto che troppi thread creano problemi di latenza dovuti al cambio di contesto e all'utilizzo delle risorse di elaborazione. È inoltre necessario aumentare il numero massimo di connessioni al pool in `botocore` modo che tutti i thread possano eseguire il download dell'oggetto S3 contemporaneamente.

Il codice di esempio utilizza un oggetto da 8,3 KB, con dati JSON, in un bucket S3. L'oggetto viene letto più volte. Dopo che la funzione Lambda ha letto l'oggetto, i dati JSON vengono decodificati in un oggetto Python. Nel dicembre 2024, il risultato dopo l'esecuzione di questo esempio è stato di 1.000 letture elaborate in 2,3 secondi e 10.000 letture elaborate in 27 secondi utilizzando una funzione Lambda configurata con 2.304 MB di memoria. AWS Lambda supporta configurazioni di memoria da 128 MB a 10.240 MB (10 GB), tuttavia l'aumento della memoria Lambda oltre 2.304 MB non ha contribuito a ridurre il tempo di esecuzione di questa particolare attività legata all'I/O.

Lo strumento AWS Lambda Power Tuning è stato utilizzato per testare diverse configurazioni di memoria Lambda e verificare il rapporto performance-to-cost migliore per l'attività. Per i risultati dei test, consulta la sezione Informazioni aggiuntive.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Competenza nello sviluppo di Python

Limitazioni

- Una funzione Lambda può avere al massimo 1.024 processi o thread di esecuzione.
- Account AWS I nuovi hanno un limite di memoria Lambda di 3.008 MB. Regola lo strumento AWS Lambda Power Tuning di conseguenza. Per ulteriori informazioni, consulta la sezione Risoluzione dei problemi.
- Amazon S3 ha un limite di 5.500 GET/HEAD richieste al secondo per prefisso partizionato.

Versioni del prodotto

- Python 3.9 o versioni successive
- AWS Cloud Development Kit (AWS CDK) v2
- AWS Command Line Interface (AWS CLI) versione 2
- AWS Lambda Power Tuning 4.3.6 (opzionale)

Architecture

Stack tecnologico Target

- AWS Lambda
- Simple Storage Service (Amazon S3)
- AWS Step Functions (se è AWS Lambda installato Power Tuning)

Architettura di destinazione

Il diagramma seguente mostra una funzione Lambda che legge gli oggetti da un bucket S3 in parallelo. Il diagramma presenta anche un flusso di lavoro Step Functions per lo strumento AWS Lambda Power Tuning per ottimizzare la memoria delle funzioni Lambda. Questa messa a punto aiuta a raggiungere un buon equilibrio tra costi e prestazioni.

Automazione e scalabilità

Le funzioni Lambda si scalano rapidamente quando necessario. Per evitare errori 503 Slow Down da Amazon S3 in caso di forte domanda, consigliamo di porre alcuni limiti alla scalabilità.

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK) v2 è un framework di sviluppo software che consente di definire e fornire Cloud AWS l'infrastruttura in codice. L'infrastruttura di esempio è stata creata per essere implementata con. AWS CDK
- AWS Command Line InterfaceAWS CLIè uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando. In questo modello, la AWS CLI versione 2 viene utilizzata per caricare un file JSON di esempio.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Simple Storage Service Amazon S3 è un servizio di storage di oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Step Functionsè un servizio di orchestrazione serverless che ti aiuta a combinare AWS Lambda funzioni e altri servizi AWS per creare applicazioni aziendali critiche.

Altri strumenti

- Python è un linguaggio di programmazione per computer generico. Il riutilizzo dei thread di lavoro inattivi è stato introdotto nella versione 3.8 di Python e il codice della funzione Lambda in questo modello è stato creato per Python versione 3.9 e successive.

Archivio di codice

Il codice per questo pattern è disponibile nel aws-lambda-parallel-download GitHub repository.

Best practice

- Questo AWS CDK costrutto si basa sulle autorizzazioni utente Account AWS dell'utente per implementare l'infrastruttura. Se prevedi di utilizzare AWS CDK Pipelines o distribuzioni tra account, consulta i sintetizzatori Stack.
- Questa applicazione di esempio non ha i log di accesso abilitati nel bucket S3. È consigliabile abilitare i log di accesso nel codice di produzione.

Epiche

Prepara l'ambiente di sviluppo

Operazione	Description	Competenze richieste
Controlla la versione installata di Python.	Questo codice è stato testato specificamente su Python 3.9 e Python 3.13 e dovrebbe funzionare su tutte le versioni tra queste versioni. Per verificare la tua versione di Python, esegui `python3 -V` nel tuo terminale e installa una versione più recente, se necessario. Per verificare che i moduli richiesti siano installat i, esegui. `python3 -c "import pip, venv"` Nessun messaggio di errore indica che i moduli sono installati correttamente e che sei pronto per eseguire questo esempio.	Architetto del cloud
Installa AWS CDK.	Per installare il, AWS CDK se non è già installato, segui le	Architetto del cloud

Operazione	Description	Competenze richieste
	istruzioni in Guida introduttiva a AWS CDK. Per confermare che la AWS CDK versione installata è 2.0 o successiva, eseguicdk -version.	
Avvia il tuo ambiente.	Per avviare il tuo ambiente, se non l'hai già fatto, segui le istruzioni riportate in Bootstrap your environment per l'utilizzo con. AWS CDK	Architetto del cloud

Clona il repository di esempio

Operazione	Description	Competenze richieste
Clonare il repository.	Per clonare l'ultima versione del repository, esegui il seguente comando: `git clone --depth 1 --branch v1.2.0 \ git@github.com:aws-samples/aws-lambda-parallel-download.git`	Architetto del cloud
Cambia la directory di lavoro nel repository clonato.	Esegui il comando seguente: `cd aws-lambda-parallel-download`	Architetto del cloud
Crea l'ambiente virtuale Python.	Per creare un ambiente virtuale Python, esegui il seguente comando:	Architetto cloud

Operazione	Description	Competenze richieste
	`python3 -m venv .venv`	
Attiva l'ambiente virtuale.	Per attivare l'ambiente virtuale, esegui il seguente comando: `source .venv/bin/activate`	Architetto del cloud
Installa le dipendenze.	Per installare le dipendenze Python, esegui il comando: `pip` `pip install -r requirements.txt`	Architetto del cloud
Sfoglia il codice.	(Facoltativo) Il codice di esempio che scarica un oggetto dal bucket S3 si trova in. `resources/parallel.py` Il codice dell'infrastruttura si trova nella cartella`parallel_download` .	Architetto del cloud

Implementa e testa l'app

Operazione	Description	Competenze richieste
Distribuire l'app.	Esegui `cdk deploy`. Annota gli AWS CDK output:	Architetto del cloud

Operazione	Description	Competenze richieste
	• `ParallelDownloadStack.LambdaFunctionARN` • `ParallelDownloadStack.SampleS3BucketName` • `ParallelDownloadStack.StateMachineARN`	
Carica un file JSON di esempio.	Il repository contiene un file JSON di esempio di circa 9 KB. Per caricare il file nel bucket S3 dello stack creato, esegui il comando seguente: `aws s3 cp sample.json s3://<ParallelDownloadStack.SampleS3BucketName>` Sostituisci `<ParallelDownloadStack.SampleS3BucketName>` con il valore corrispondente dall'output. AWS CDK	Architetto del cloud

Operazione	Description	Competenze richieste
Esegui l'app.	Per eseguire l'app, procedi come segue: 1. Accedi a Console di gestione AWS, vai alla console Lambda e individua la funzione Lambda che contiene l'ARN dall'output. AWS CDK `ParallelDownloadStack.LambdaFunctionARN` 2. Nella scheda Test, modifica il codice JSON dell'evento nel modo seguente: `{"objectKey": "sample.json"}` 3. Scegli Test (Esegui test). 4. Per vedere il risultato, scegli dettagli. I dettagli mostreranno le statistiche del download parallelo, le informazioni sull'esecuzione e i log.	Architetto del cloud
Aggiungi il numero di download.	(Facoltativo) Per eseguire 1.500 chiamate get object, utilizzate il seguente codice JSON in Event JSON del parametro: `Test` `{"repeat": 1500, "objectKey": "sample.json"}`	Architetto del cloud

Opzionale: esegui AWS Lambda Power Tuning

Operazione	Description	Competenze richieste
Esegui lo strumento AWS Lambda Power Tuning.	1. Accedi alla console e vai a Step Functions. 2. Individua la macchina a stati con l'ARN dall' AWS CDK output. `ParallelDownloadStack.StateMachineARN` 3. Scegliete Avvia esecuzione e incollate il seguente codice JSON: `{ "lambdaARN": "<ParallelDownloadStack.LambdaFunctionARN>", "num": 10, "strategy": "balanced", "payload": {"repeat": 2000, "objectKey": "sample.json"} }` Ricordati di sostituirlo `<ParallelDownloadStack.LambdaFunctionARN>` con il valore dell' AWS CDK output. Alla fine dell'esecuzione, il risultato sarà visualizzato nella	Architetto del cloud

Operazione	Description	Competenze richieste
	scheda Esecuzione input e output.	
Visualizza i risultati del AWS Lambda Power Tuning in un grafico.	Nella scheda Execution input and output, copiate il link della `visualization` proprietà e incollatelo in una nuova scheda del browser.	Architetto del cloud

Eliminazione

Operazione	Description	Competenze richieste
Rimuovi gli oggetti dal bucket S3.	Prima di distruggere le risorse distribuite, rimuovi tutti gli oggetti dal bucket S3: `aws s3 rm s3://<ParallelDownloadStack.SampleS3BucketName> \ --recursive` Ricordati di sostituirlo `<ParallelDownloadStack.SampleS3BucketName>` con il valore delle uscite. AWS CDK	Architetto del cloud
Distruggi le risorse.	Per distruggere tutte le risorse create per questo programma pilota, esegui il seguente comando: `cdk destroy`	Architetto del cloud

risoluzione dei problemi

Problema	Soluzione
`'MemorySize' value failed to satisfy constraint: Member must have value less than or equal to 3008`	Per i nuovi account, potresti non essere in grado di configurare più di 3.008 MB nelle tue funzioni Lambda. Per testare l'utilizzo AWS Lambda di Power Tuning, aggiungi la seguente proprietà all'input JSON quando avvii l'esecuzione di Step Functions: `"powerValues": [ 512, 1024, 1536, 2048, 2560, 3008 ]`

Risorse correlate

- Python — concurrent.futures. ThreadPoolExecutor
- Quote Lambda: configurazione, distribuzione ed esecuzione delle funzioni
- Lavorare con il AWS CDK in Python
- Funzioni di profilazione con AWS Lambda Power Tuning

Informazioni aggiuntive

Codice

Il seguente frammento di codice esegue l'elaborazione I/O parallela:

```
with ThreadPoolExecutor(max_workers=MAX_WORKERS) as executor:
  for result in executor.map(a_function, (the_arguments)):
    ...
```

`ThreadPoolExecutor`Riutilizza i thread quando diventano disponibili.

Test e risultati

Questi test sono stati condotti nel dicembre 2024.

Il primo test ha elaborato 2.500 letture di oggetti, con il seguente risultato.

A partire da 3.009 MB, il livello di tempo di elaborazione è rimasto pressoché invariato per ogni aumento di memoria, ma il costo è aumentato all'aumentare delle dimensioni della memoria.

Un altro test ha analizzato l'intervallo tra 1.536 MB e 3.072 MB di memoria, utilizzando valori multipli di 256 MB ed elaborando 10.000 letture di oggetti, con i seguenti risultati.

Il performance-to-cost rapporto migliore è stato ottenuto con la configurazione Lambda da 2.304 MB di memoria.

A titolo di confronto, un processo sequenziale di 2.500 letture di oggetti ha richiesto 47 secondi. Il processo parallelo che utilizza la configurazione Lambda da 2.304 MB ha richiesto 7 secondi, ovvero l'85% in meno.

Invia dati di telemetria da AWS Lambda a OpenSearch per analisi e visualizzazione in tempo reale

Tabby Ward, Guy Bachar e David Kilzer, Amazon Web Services

Riepilogo

Le applicazioni moderne stanno diventando sempre più distribuite e basate sugli eventi, il che rafforza la necessità di monitoraggio e osservabilità in tempo reale. AWS Lambda è un servizio di elaborazione serverless che svolge un ruolo cruciale nella creazione di architetture scalabili e basate sugli eventi. Tuttavia, il monitoraggio e la risoluzione dei problemi delle funzioni Lambda possono essere difficili se ci si affida esclusivamente ad Amazon CloudWatch Logs, che può introdurre latenza e periodi di conservazione limitati.

Per affrontare questa sfida, AWS ha introdotto l'API Lambda Telemetry, che consente alle funzioni Lambda di inviare dati di telemetria direttamente a strumenti di monitoraggio e osservabilità di terze parti. Questa API supporta lo streaming in tempo reale di log, metriche e tracce e fornisce una visione completa e tempestiva delle prestazioni e dello stato delle funzioni Lambda.

Questo modello spiega come integrare l'API Lambda Telemetry con OpenSearch, un motore di ricerca e analisi distribuito open source. OpenSearch offre una piattaforma potente e scalabile per l'acquisizione, l'archiviazione e l'analisi di grandi volumi di dati, che la rende la scelta ideale per i dati di telemetria Lambda. In particolare, questo modello dimostra come inviare i log da una funzione Lambda scritta in Python direttamente a un cluster OpenSearch utilizzando un'estensione Lambda fornita da. AWS Questa soluzione è flessibile e personalizzabile, quindi puoi creare la tua estensione Lambda o modificare il codice sorgente di esempio per modificare il formato di output come desideri.

Il modello spiega come impostare e configurare l'integrazione dell'API Lambda Telemetry e include le migliori pratiche per la sicurezza OpenSearch, l'ottimizzazione dei costi e la scalabilità. L'obiettivo è aiutarti a ottenere informazioni più approfondite sulle tue funzioni Lambda e migliorare l'osservabilità complessiva delle tue applicazioni serverless.

> Nota: questo modello si concentra sull'integrazione dell'API di telemetria Lambda con managed. OpenSearch Tuttavia, i principi e le tecniche discussi sono applicabili anche all'autogestione e a Elasticsearch. OpenSearch

Prerequisiti e limitazioni

Prima di iniziare il processo di integrazione, assicurati di avere i seguenti prerequisiti:

Account AWS: Un attivo Account AWS con le autorizzazioni appropriate per creare e gestire le seguenti risorse: AWS

- AWS Lambda
- AWS Identity and Access Management (IAM)
- Amazon OpenSearch Service (se utilizzi un OpenSearch cluster gestito)

OpenSearch cluster:

- È possibile utilizzare un OpenSearch cluster autogestito esistente o un servizio gestito come OpenSearch Service.
- Se utilizzi OpenSearch Service, configura il OpenSearch cluster seguendo le istruzioni in Getting started with Amazon OpenSearch Service nella documentazione del OpenSearch Servizio.
- Assicurati che il OpenSearch cluster sia accessibile dalla tua funzione Lambda e sia configurato con le impostazioni di sicurezza necessarie, come le politiche di accesso, la crittografia e l'autenticazione.
- Configura il OpenSearch cluster con le mappature degli indici e le impostazioni necessarie per importare i dati di telemetria Lambda. Per ulteriori informazioni, consulta Caricamento di dati di streaming in Amazon OpenSearch Service nella documentazione del OpenSearch servizio.

Connettività di rete:

- Assicurati che la tua funzione Lambda disponga della connettività di rete necessaria per accedere al OpenSearch cluster. Per indicazioni su come configurare le impostazioni del cloud privato virtuale (VPC), consulta Avvio dei domini Amazon OpenSearch Service all'interno di un VPC nella documentazione del servizio. OpenSearch

Ruoli e politiche IAM:

- Crea un ruolo IAM con le autorizzazioni necessarie affinché la tua funzione Lambda possa accedere al cluster e accedere OpenSearch alle tue credenziali archiviate in. Gestione dei segreti AWS

- Allega le policy IAM appropriate al ruolo, ad esempio la `AWSLambdaBasicExecutionRole` policy e le eventuali autorizzazioni aggiuntive necessarie per interagire con. OpenSearch
- Verifica che le autorizzazioni IAM concesse alla tua funzione Lambda consentano di scrivere OpenSearch dati nel cluster. Per informazioni sulla gestione delle autorizzazioni IAM, consulta Definizione delle autorizzazioni della funzione Lambda con un ruolo di esecuzione nella documentazione Lambda.

Conoscenza del linguaggio di programmazione:

- Avrai bisogno di una conoscenza di base di Python (o del linguaggio di programmazione di tua scelta) per comprendere e modificare il codice di esempio per la funzione Lambda e l'estensione Lambda.

Ambiente di sviluppo:

- Configura un ambiente di sviluppo locale con gli strumenti e le dipendenze necessari per creare e distribuire funzioni ed estensioni Lambda.

AWS CLI oppure: Console di gestione AWS

- Installa e configura il AWS Command Line Interface (AWS CLI) o usalo Console di gestione AWS con le credenziali appropriate per interagire con quello richiesto. Servizi AWS

Monitoraggio e registrazione:

- Acquisisci familiarità con le migliori pratiche di monitoraggio e registrazione su AWS servizi come Amazon CloudWatch e AWS CloudTrail per scopi di monitoraggio e controllo.
- Controlla CloudWatch i log per la tua funzione Lambda per identificare eventuali errori o eccezioni relativi all'integrazione dell'API Lambda Telemetry. Per una guida alla risoluzione dei problemi, consulta la documentazione dell'API Lambda Telemetry.

Architecture

Questo modello utilizza OpenSearch Service per archiviare registri e dati di telemetria generati dalle funzioni Lambda. Questo approccio consente di trasmettere rapidamente i log direttamente al

OpenSearch cluster, riducendo la latenza e i costi associati all'utilizzo di Logs come intermediario. CloudWatch

> **Note**
>
> Il codice di estensione Lambda può inviare telemetria a OpenSearch Service, utilizzando direttamente l' OpenSearch API o utilizzando una libreria client. OpenSearch L'estensione Lambda può utilizzare le operazioni di massa supportate dall' OpenSearch API per raggruppare gli eventi di telemetria in batch e inviarli a OpenSearch Service in un'unica richiesta.

Il seguente diagramma del flusso di lavoro illustra il flusso di lavoro di registro per le funzioni Lambda quando si utilizza un OpenSearch cluster come endpoint.

L'architettura include i seguenti componenti:

- Funzione Lambda: la funzione serverless che genera log e dati di telemetria durante l'esecuzione.
- Estensione Lambda: un'estensione basata su Python che utilizza l'API Lambda Telemetry per l'integrazione diretta con il cluster. OpenSearch Questa estensione viene eseguita insieme alla funzione Lambda nello stesso ambiente di esecuzione.
- API di telemetria Lambda: l'API che consente alle estensioni Lambda di inviare dati di telemetria, inclusi log, metriche e tracce, direttamente a strumenti di monitoraggio e osservabilità di terze parti.
- Cluster Amazon OpenSearch Service: un OpenSearch cluster gestito ospitato su AWS. Questo cluster è responsabile dell'acquisizione, dell'archiviazione e dell'indicizzazione dei dati di registro trasmessi dalla funzione Lambda tramite l'estensione Lambda.

Il flusso di lavoro prevede i seguenti passaggi:

1. La funzione Lambda viene chiamata e genera log e dati di telemetria durante la sua esecuzione.
2. L'estensione Lambda funziona insieme alla funzione per acquisire i log e i dati di telemetria utilizzando l'API Lambda Telemetry.
3. L'estensione Lambda stabilisce una connessione sicura con il cluster di OpenSearch servizi e trasmette i dati di registro in tempo reale.

4. Il cluster di OpenSearch servizi inserisce, indicizza e archivia i dati di registro per renderli disponibili per la ricerca, l'analisi e la visualizzazione tramite l'uso di strumenti come Kibana o altre applicazioni compatibili.

Evitando CloudWatch i log e inviando i dati di log direttamente al cluster, questa soluzione offre diversi vantaggi: OpenSearch

- Streaming e analisi dei log in tempo reale, che consentono una risoluzione dei problemi più rapida e una migliore osservabilità.
- Latenza ridotta e potenziali limitazioni di conservazione associate CloudWatch ai log.
- Flessibilità di personalizzare l'estensione Lambda o creare un'estensione personalizzata per formati di output specifici o elaborazioni aggiuntive.
- Integrazione con le funzionalità di ricerca, analisi e visualizzazione di OpenSearch Service per l'analisi e il monitoraggio dei log.

La sezione Epics fornisce step-by-step istruzioni per configurare l'estensione Lambda, configurare la funzione Lambda e integrare con il cluster di servizi. OpenSearch Per considerazioni sulla sicurezza, strategie di ottimizzazione dei costi e suggerimenti per il monitoraggio e la risoluzione dei problemi della soluzione, consulta la sezione Best practice.

Tools (Strumenti)

Servizi AWS

- AWS Lambda è un servizio di elaborazione che consente di eseguire del codice senza la necessità di effettuare il provisioning o la gestione dei server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo.
- Amazon OpenSearch Service è un servizio completamente gestito fornito da AWS che semplifica la distribuzione, il funzionamento e la scalabilità OpenSearch dei cluster nel cloud.
- Le estensioni Lambda estendono la funzionalità delle funzioni Lambda eseguendo codice personalizzato insieme a esse. Puoi utilizzare le estensioni Lambda per integrare Lambda con vari strumenti di monitoraggio, osservabilità, sicurezza e governance.
- AWS Lambda L'API di telemetria ti consente di utilizzare le estensioni per acquisire dati avanzati di monitoraggio e osservabilità direttamente da Lambda e inviarli a una destinazione di tua scelta.
- CloudFormationti aiuta a modellare e configurare AWS le tue risorse in modo da dedicare meno tempo alla gestione di tali risorse e più tempo a concentrarti sulle tue applicazioni.

Archivi di codice

- AWS Lambda Extensions include demo e progetti di AWS esempio di AWS partner per aiutarti a iniziare a creare le tue estensioni.
- Example Lambda Telemetry Integrations for fornisce OpenSearch un'estensione Lambda di esempio che dimostra come inviare i log da una funzione Lambda a un cluster. OpenSearch

Altri strumenti

- OpenSearchè un motore di ricerca e analisi distribuito open source che fornisce una potente piattaforma per l'acquisizione, l'archiviazione e l'analisi di grandi volumi di dati.
- Kibana è uno strumento di visualizzazione ed esplorazione dei dati open source che puoi utilizzare con. OpenSearch Nota che l'implementazione della visualizzazione e dell'analisi esula dall'ambito di questo modello. Per ulteriori informazioni, consulta la documentazione di Kibana e altre risorse.

Best practice

Quando integri l'API di telemetria Lambda con OpenSearch, prendi in considerazione le seguenti best practice.

Sicurezza e controllo degli accessi

- Comunicazione sicura: crittografa tutte le comunicazioni tra le funzioni Lambda e OpenSearch il cluster utilizzando HTTPS. Configura le SSL/TLS impostazioni necessarie nell'estensione e OpenSearch nella configurazione Lambda.
- Autorizzazioni IAM:
  - Le estensioni vengono eseguite nello stesso ambiente di esecuzione della funzione Lambda, quindi ereditano lo stesso livello di accesso a risorse come il file system, la rete e le variabili di ambiente.
  - Concedi le autorizzazioni IAM minime necessarie alle tue funzioni Lambda per accedere all'API Lambda Telemetry e scrivere dati nel cluster. OpenSearch Utilizza il principio del privilegio minimo per limitare l'ambito delle autorizzazioni.
- OpenSearch controllo degli accessi: implementa un controllo granulare degli accessi nel OpenSearch cluster per limitare l'accesso ai dati sensibili. Utilizza le funzionalità di sicurezza integrate, come l'autenticazione degli utenti, il controllo degli accessi basato sui ruoli e le autorizzazioni a livello di indice, in. OpenSearch

- Estensioni affidabili: installa sempre le estensioni solo da una fonte attendibile. Utilizza strumenti Infrastructure as Code (IaC) CloudFormation per semplificare il processo di associazione della stessa configurazione di estensione, incluse le autorizzazioni IAM, a più funzioni Lambda. Gli strumenti IAc forniscono anche un registro di controllo delle estensioni e delle versioni utilizzate in precedenza.
- Gestione dei dati sensibili: quando crei estensioni, evita di registrare dati sensibili. Pulisci i payload e i metadati prima di registrarli o renderli persistenti per scopi di controllo.

Ottimizzazione dei costi

- Monitoraggio e avviso: configura meccanismi di monitoraggio e avviso per tenere traccia del volume di dati inviati dalle tue funzioni OpenSearch Lambda. Questo ti aiuterà a identificare e risolvere eventuali sforamenti dei costi.
- Conservazione dei dati: valuta attentamente il periodo di conservazione dei dati appropriato per i tuoi dati di telemetria Lambda in. OpenSearch Periodi di conservazione più lunghi possono aumentare i costi di archiviazione, quindi bilancia le esigenze di osservabilità con l'ottimizzazione dei costi.
- Compressione e indicizzazione: abilita la compressione dei dati e ottimizza la tua strategia di OpenSearch indicizzazione per ridurre l'ingombro di archiviazione dei dati di telemetria Lambda.
- Riduzione della dipendenza da CloudWatch: integrando direttamente l'API di telemetria Lambda con OpenSearch, è possibile ridurre potenzialmente la dipendenza dai CloudWatch log, con conseguenti risparmi sui costi. Questo perché l'API Lambda Telemetry consente di inviare i log direttamente a OpenSearch, evitando così la necessità di archiviare ed elaborare i dati. CloudWatch

Scalabilità e affidabilità

- Elaborazione asincrona: utilizza modelli di elaborazione asincroni, come Amazon Simple Queue Service (Amazon SQS) o Amazon Kinesis, per disaccoppiare l'esecuzione della funzione Lambda dall'ingestione dei dati. OpenSearch Questo aiuta a mantenere la reattività delle funzioni Lambda e migliora l'affidabilità complessiva del sistema.
- OpenSearch scalabilità del cluster: monitora le prestazioni e l'utilizzo delle risorse del OpenSearch cluster e ridimensionalo verso l'alto o verso il basso secondo necessità per gestire il volume crescente di dati di telemetria Lambda.

- Failover e disaster recovery: implementa una solida strategia di disaster recovery per il OpenSearch cluster, che include backup regolari e la capacità di ripristinare rapidamente i dati in caso di guasto.

Osservabilità e monitoraggio

- Dashboard e visualizzazioni: utilizza Kibana o altri strumenti di dashboard per creare dashboard e visualizzazioni personalizzate che forniscono informazioni sulle prestazioni e sullo stato delle funzioni Lambda in base ai dati di telemetria inclusi. OpenSearch
- Avvisi e notifiche: configura avvisi e notifiche per monitorare in modo proattivo anomalie, errori o problemi di prestazioni nelle funzioni Lambda. Integra questi avvisi e notifiche con i processi di gestione degli incidenti esistenti.
- Tracciamento e correlazione: assicurati che i dati di telemetria Lambda includano informazioni di tracciamento pertinenti, come richieste IDs o correlazioni, per consentire end-to-end l'osservabilità e la risoluzione dei problemi tra le applicazioni IDs serverless distribuite.

Seguendo queste best practice, puoi assicurarti che l'integrazione dell'API Lambda Telemetry con OpenSearch sia sicura, economica e scalabile e fornisca un'osservabilità completa per le tue applicazioni serverless.

Epiche

Crea e distribuisci il livello di estensione Lambda

Operazione	Description	Competenze richieste
Scarica il codice sorgente.	Scarica le estensioni di esempio dal repository AWS Lambda Extensions.	Sviluppatore di app, architetto cloud
Accedi alla cartella `python-example-telemetry-opensearch-extension` .	L'archivio delle AWS Lambda estensioni che hai scaricato contiene numerosi esempi per diversi casi d'uso e runtime linguistici. Vai alla cartella python-example-telemetry-opensearch-extension per	Sviluppatore di app, architetto cloud

Operazione	Description	Competenze richieste
	usare l'estensione OpenSearch Python, a cui invia i log. OpenSearch	
Aggiungi le autorizzazioni per eseguire l'endpoint di estensione.	Esegui il comando seguente per rendere eseguibile l'endpoint dell'estensione: `chmod +x python-example-telemetry-opensearch-extension/extension.py`	Sviluppatore di app, architetto cloud
Installa le dipendenze dell'estensione localmente.	Esegui il seguente comando per installare le dipendenze locali per il codice Python: `pip3 install -r python-example-telemetry-opensearch-extension/requirements.txt -t ./python-example-telemetry-opensearch-extension/` Queste dipendenze verranno montate insieme al codice di estensione.	Sviluppatore di app, architetto cloud

Operazione	Description	Competenze richieste

Crea un pacchetto.zip per l'estensione per distribuirla come livello.	Il file con estensione zip deve contenere una directory principale denominat `aextensions/` , in cui si trova l'eseguibile dell'este nsione, e un'altra directory principale chiamata`python-example-telemetry-opensearch-extension/` , in cui si trovano la logica di base dell'estensione e le sue dipendenze.
	
	Crea il pacchetto.zip per l'estensione:
	
	`chmod +x extensions/python-example-telemetry-opensearch-extension`
	`zip -r extension.zip extensions python-example-telemetry-opensearch-extension`
		Sviluppatore di app, architetto cloud

Operazione	Description	Competenze richieste
Implementa l'estensione come livello Lambda.	Pubblicate il layer utilizzando il file di estensione .zip e il seguente comando: `aws lambda publish-layer-version \ --layer-name "python-example-telemetry-opensearch-extension" \ --zip-file "fileb://extension.zip"`	Sviluppatore di app, architetto cloud

Integra l'estensione nella tua funzione

Operazione	Description	Competenze richieste
Aggiungi il livello alla tua funzione.	1. Accedi Console di gestione AWS e apri la pagina Funzioni della AWS Lambda console. 2. Seleziona la tua funzione. 3. In Livelli, scegliere Aggiungi un livello. 4. In Scegli un livello, scegli Livelli personalizzati come sorgente del livello e aggiungi il tuo livello. Per ulteriori informazioni sull'aggiunta di un layer alla funzione Lambda, consulta la documentazione Lambda.	Sviluppatore di app, architetto cloud

Operazione	Description	Competenze richieste
Imposta le variabili di ambiente per la funzione.	Nella pagina della funzione, scegliete la scheda Configura zione e aggiungete le seguenti variabili di ambiente alla funzione: • `URL`— L'URI dell' OpenSearch endpoint a cui verranno inviati i log. • `AUTH_SECRET` — L'ARN delle OpenSearch credenzia li memorizzate in. Gestione dei segreti AWS Questo dovrebbe essere memorizza to come coppia chiave-va lore e avere due chiavi: e. `username` `password` • `PLATFORM_ INDEX` `FUNCTION_ INDEX` , e `EXTENSION _INDEX` — I nomi degli indici in cui verranno archiviati i dati di telemetri a, i registri delle funzioni e i registri delle estension i. Assicurati che rispettino i criteri di denominazione corretti. Altrimenti, i tuoi indici non verranno creati. • `DISPATCH_MIN_BATCH _SIZE` — Il numero di eventi di registro che si desidera raggruppare. Tuttavia, quando la funzione	Sviluppatore di app, architetto cloud

Operazione	Description	Competenze richieste
	si spegne, i log verranno inviati indipendentemente da questa impostazione.	

Aggiungi istruzioni di registrazione e verifica la tua funzione

Operazione	Description	Competenze richieste

Aggiungi istruzioni di registrazione alla tua funzione.	Aggiungi istruzioni di registrazione alla tua funzione utilizzando uno dei meccanismi di registrazione integrati o il modulo di registrazione preferito.
	
	Ecco alcuni esempi di registrazione dei messaggi in Python:
	
	`print("Your Log Message Here")`
	`logger = logging.getLogger(__name__)`
	
	`logger.info("Test Info Log.")`
	`logger.error("Test Error Log.")`
		Sviluppatore di app, architetto cloud

Prova la tua funzione .	1. Nella pagina delle funzioni, seleziona la scheda Test . 2. Crea un evento di test per la tua funzione ed esegui il test. Per ulteriori informazioni, consulta Testare le funzioni Lambda nella	Sviluppatore di app, architetto cloud

Operazione	Description	Competenze richieste
	console nella documentazione di Lambda. Dovresti vedere Esecuzione della funzione: riuscita se tutto funziona correttamente.	

Visualizza i tuoi dati di accesso OpenSearch

Operazione	Description	Competenze richieste
Interroga i tuoi indici.	In OpenSearch, esegui il comando seguente per interrogare i tuoi indici: `SELECT * FROM index-name` I log dovrebbero essere visualizzati nei risultati della query.	Architetto del cloud

risoluzione dei problemi

Problema	Soluzione
Problemi di connettività	• Verifica che la tua funzione Lambda disponga della connettività di rete necessari a per accedere al OpenSearch cluster. Consulta la documentazione del OpenSearch servizio per indicazioni sulla configurazione delle impostazioni VPC.

Problema	Soluzione
	• Verifica che le autorizzazioni IAM concesse alla tua funzione Lambda consentano di scrivere OpenSearch dati nel cluster. Consulta la documentazione Lambda per informazioni sulla gestione delle autorizza zioni IAM.
Errori di inserimento dei dati	• Controlla CloudWatch i log per la tua funzione Lambda per identificare eventuali errori o eccezioni relativi all'integrazione dell'API Lambda Telemetry. Consulta la documentazione dell'API Lambda Telemetry per una guida alla risoluzione dei problemi. • Verifica che il OpenSearch cluster sia configurato correttamente e disponga delle mappature degli indici e delle impostazioni necessarie per importare i dati di telemetri a Lambda. Consulta la documentazione per ulteriori informazioni. OpenSearch

Risorse correlate

- Esempi di integrazioni di telemetria Lambda per (repository) OpenSearch GitHub
- Potenzia le funzioni Lambda utilizzando le estensioni Lambda (documentazione Lambda)
- API di telemetria Lambda (documentazione Lambda)
- Presentazione dell'API di AWS Lambda telemetria (post sul blog)AWS
- Integrazione dell' AWS Lambda API di telemetria con Prometheus e (post sul blog) OpenSearch AWS

Informazioni aggiuntive

Modifica della struttura dei log

Per impostazione predefinita, l'estensione invia i log come documento annidato a OpenSearch . Ciò consente di eseguire interrogazioni annidate per recuperare i valori delle singole colonne.

Se l'output di registro predefinito non soddisfa le tue esigenze specifiche, puoi personalizzarlo modificando il codice sorgente dell'estensione Lambda fornita da. AWS AWS incoraggia i clienti ad adattare l'output alle proprie esigenze aziendali. Per modificare l'output del registro, individua la `dispatch_to_opensearch` funzione nel `telemetry_dispatcher.py` file all'interno del codice sorgente dell'estensione e apporta le modifiche necessarie.

Configura un router cellulare serverless per un'architettura basata su celle

Mian Tariq e Ioannis Lioupras, Amazon Web Services

Riepilogo

In quanto punto di accesso a un sistema globale di applicazioni basate su celle, il router cellulare è responsabile dell'assegnazione efficiente degli utenti alle celle appropriate e della fornitura degli endpoint agli utenti. Il router cellulare gestisce funzioni come la memorizzazione delle user-to-cell mappature, il monitoraggio della capacità delle celle e la richiesta di nuove celle quando necessario. È importante mantenere la funzionalità del router cellulare durante potenziali interruzioni.

Il framework di progettazione cell-router in questo modello si concentra su resilienza, scalabilità e ottimizzazione complessiva delle prestazioni. Il modello utilizza il routing statico, in cui i client memorizzano nella cache gli endpoint al momento dell'accesso iniziale e comunicano direttamente con le celle. Questo disaccoppiamento migliora la resilienza del sistema contribuendo a garantire la funzionalità ininterrotta dell'applicazione basata su celle in caso di problemi tra cellulare e router.

Questo modello AWS CloudFormation utilizza un modello per implementare l'architettura. Per informazioni dettagliate su ciò che viene distribuito dal modello o per distribuire la stessa configurazione utilizzando il Console di gestione AWS, consulta la sezione Informazioni aggiuntive.

> **Important**
>
> La dimostrazione, il codice e il CloudFormation modello presentati in questo modello hanno solo scopo esplicativo. Il materiale fornito ha il solo scopo di illustrare il modello di progettazione e di facilitarne la comprensione. La demo e il codice non sono pronti per la produzione e non devono essere utilizzati per attività di produzione dal vivo. Qualsiasi tentativo di utilizzare il codice o la demo in un ambiente di produzione è fortemente sconsigliato ed è a rischio dell'utente. Consigliamo di consultare i professionisti appropriati e di eseguire test approfonditi prima di implementare questo modello o uno qualsiasi dei suoi componenti in un ambiente di produzione.

Prerequisiti e limitazioni

Prerequisiti

- Un account Amazon Web Services (AWS) attivo

- L'ultima versione di AWS Command Line Interface (AWS CLI)
- Credenziali AWS con le autorizzazioni necessarie per creare lo CloudFormation stack, AWS Lambda le funzioni e le risorse correlate

Versioni del prodotto

- Python 3.12

Architecture

Il diagramma seguente mostra un design di alto livello del router cellulare.

Il diagramma illustra il seguente flusso di lavoro:

1. L'utente contatta Amazon API Gateway, che funge da interfaccia per gli endpoint API cell-router.
2. Amazon Cognito gestisce l'autenticazione e l'autorizzazione.
3. Il AWS Step Functions flusso di lavoro è composto dai seguenti componenti:
   - Orchestrator – Gli `Orchestrator` usi AWS Step Functions per creare un flusso di lavoro o una macchina a stati. Il flusso di lavoro viene attivato dall'API del router cellulare. `Orchestrator`Esegue le funzioni Lambda in base al percorso della risorsa.
   - Dispatcher – La funzione `Dispatcher` Lambda identifica e assegna una cella statica per ogni nuovo utente registrato. La funzione cerca la cella con il minor numero di utenti, la assegna all'utente e restituisce gli endpoint.
   - Mapper – L'`Mapper`operazione gestisce le user-to-cell mappature all'interno del database `RoutingDB` Amazon DynamoDB creato dal modello. CloudFormation Quando viene attivata, la `Mapper` funzione fornisce agli utenti già assegnati i propri endpoint.
   - Scaler – La `Scaler` funzione tiene traccia dell'occupazione delle celle e della capacità disponibile. Se necessario, la `Scaler` funzione può inviare una richiesta tramite Amazon Simple Queue Service (Amazon SQS) al livello Provision and Deploy per richiedere nuove celle.
   - Validator – La `Validator` funzione convalida gli endpoint delle celle e rileva eventuali problemi.
4. `RoutingDB`Memorizza le informazioni e gli attributi delle celle (endpoint API, stato Regione AWS, metriche).
5. Quando la capacità disponibile delle celle supera una soglia, il router cellulare richiede servizi di provisioning e distribuzione tramite Amazon SQS per creare nuove celle.

Quando vengono create nuove celle, `RoutingDB` viene aggiornato dal livello Provision and Deploy. Tuttavia, tale processo non rientra nell'ambito di questo modello. Per una panoramica dei presupposti di progettazione dell'architettura basata su celle e dettagli sul design cell-router utilizzato in questo modello, vedere la sezione Informazioni aggiuntive.

Tools (Strumenti)

Servizi AWS

- Amazon API Gateway ti aiuta a creare, pubblicare, gestire, monitorare e proteggere REST, HTTP e WebSocket APIs su qualsiasi scala.
- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita attraverso Account AWS e. Regioni AWS
- Amazon Cognito fornisce autenticazione, autorizzazione e gestione degli utenti per app Web e mobili.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Simple Queue Service (Amazon SQS) fornisce una coda ospitata sicura, durevole e disponibile che ti aiuta a integrare e disaccoppiare sistemi e componenti software distribuiti.
- AWS Step Functionsè un servizio di orchestrazione serverless che consente di combinare funzioni Lambda e altro Servizi AWS per creare applicazioni aziendali critiche.

Altri strumenti

- Python è un linguaggio di programmazione per computer generico.

Deposito di codice

Il codice per questo pattern è disponibile nel repository GitHub Serverless-Cell-Router.

Best practice

Per le migliori pratiche per la creazione di architetture basate su celle, consulta la seguente guida Well-Architected AWS :

- Ridurre l'ambito di impatto con l'architettura basata su celle
- AWS Well-Architected Framework Reliability Pillar REL1: 0-BP04 Utilizza architetture bulkhead per limitare l'ambito dell'impatto

Epiche

Prepara i file sorgente

Operazione	Description	Competenze richieste
Clona il repository di codice di esempio.	Per clonare il Serverless-Cell-Router GitHub repository sul tuo computer, usa il seguente comando: `git clone https://github.com/aws-samples/Serverless-Cell-Router/`	Developer
Imposta credenziali AWS CLI temporanee.	Configura il AWS CLI con le credenziali per il tuo. Account AWS Questa procedura dettagliata utilizza credenziali temporanee fornite dalla riga di comando di AWS IAM Identity Center o dall'opzione di accesso programmatico. In questo modo vengono impostate le variabili `AWS_ACCESS_KEY_ID` `AWS_SECRET_ACCESS_KEY` , e di	Developer

Operazione	Description	Competenze richieste
	`AWS_SESSION_TOKEN` AWS ambiente con le credenziali appropriate da utilizzare con. AWS CLI	
Crea un bucket S3.	Crea un bucket S3 che verrà utilizzato per archiviar e e accedere alle funzioni Serverless-Cell-Router Lambda per la distribuzione tramite il modello. CloudForm ation Per creare il bucket S3, usa il seguente comando: `aws s3api create-bucket --bucket <bucket name> --region eu-central-1 --create-bucket-configuration LocationConstraint=eu-central-1`	Developer

Operazione	Description	Competenze richieste

Crea file.zip.	Crea un file.zip per ogni funzione Lambda che si trova nella directory Functions . Questi file.zip verranno utilizzati per distribuire le funzioni Lambda. Su un Mac, usa i seguenti comandi: `zip`
	`zip -j mapper-scr.zip Functions/Mapper.py`
	`zip -j dispatcher-scr.zip Functions/Dispatcher.py`
	`zip -j scaler-scr.zip Functions/Scaler.py`
	`zip -j cp validator-scr.zip Functions/Validator.py`
	`zip -j dynamodbDummyData-scr.zip Functions/DynamodbDummyData.py`
		Developer

Operazione	Description	Competenze richieste

Copia i file.zip nel bucket S3.	Per copiare tutti i file.zip della funzione Lambda nel bucket S3, usa i seguenti comandi:
	`aws s3 cp mapper-scr.zip s3://<bucket name>`
	`aws s3 cp dispatcher-scr.zip s3://<bucket name>`
	`aws s3 cp scaler-scr.zip s3://<bucket name>`
	`aws s3 cp validator-scr.zip s3://<bucket name>`
	`aws s3 cp dynamodbDummyData-scr.zip s3://<bucket name>`
		Developer

Crea lo stack CloudFormation

Operazione	Description	Competenze richieste

Implementa il CloudFormation modello.	Per distribuire il CloudFormation modello, esegui il comando seguente: AWS CLI
	`aws cloudformation create-stack --stack-name serverless.cell-router \`
	`--template-body file://Serverless-Cell-Router-Stack-v10.yaml \`
	`--capabilities CAPABILITY_IAM \`
		Developer

Operazione	Description	Competenze richieste
	`--parameters ParameterKey=LambdaFunctionMapperS3KeyParameterSCR,ParameterValue=mapper-scr.zip \ ParameterKey=LambdaFunctionDispatcherS3KeyParameterSCR,ParameterValue=dispatcher-scr.zip \ ParameterKey=LambdaFunctionScalerS3KeyParameterSCR,ParameterValue=scaler-scr.zip \ ParameterKey=LambdaFunctionAddDynamoDBDummyItemsS3KeyParameterSCR,ParameterValue=dynamodbDummyData-scr.zip \ ParameterKey=LambdaFunctionsS3BucketParameterSCR,ParameterValue=<S3 bucket storing lambda zip files> \ ParameterKey=CognitoDomain,ParameterValue=<Cognito Domain Name> \ --region <enter your aws region id, e.g. "eu-central-1">`	

Operazione	Description	Competenze richieste
Verifica lo stato di avanzamento.	Accedi a Console di gestione AWS, apri la CloudFormation console all'indirizzo https://console.aws.amazon.com/cloudformation/e controlla lo stato di avanzamento dello sviluppo dello stack. Quando lo stato è`CREATE_COMPLETE` , lo stack è stato distribuito correttamente.	Developer

Valuta e verifica

Operazione	Description	Competenze richieste

Assegna celle all'utente.	Per avviare il`Orchestrator` , esegui il seguente comando curl:
	`curl -X POST \ -H "Authorization: Bearer {User id_token}" \ https://xxxxxx.execute-api.eu-central-1.amazonaws.com/Cell_Router_Development/cells`
	`Orchestrator` Attiva l'esecuzione della funzione. `Dispatcher` A sua volta, verifica l'esistenza dell'utente. `Dispatcher` Se l'utente viene trovato, `Dispatche`
		Developer

Operazione	Description	Competenze richieste
	`r` restituisce l'ID della cella e l'endpoint URLs associati. Se l'utente non viene trovato, gli `Dispatcher` assegna una cella e invia l'ID della cella alla `Scaler` funzione per la valutazione della capacità residua della cella assegnata. La risposta della `Scaler` funzione è la seguente: `"cellID : cell-0002 , endPoint_1 : https://xxxxx.execute-api.eu-north-1 .amazonaws.com/ , endPoint_2 : https://xxxxxxx.execute-api .eu-central-1.amaz onaws.com/"`	

Operazione	Description	Competenze richieste

Recupera le celle utente.	Per utilizzare la funzione `Orchestrator` per eseguire la `Mapper` funzione, esegui il comando seguente:
	`curl -X POST \ -H "Authorization: Bearer {User id_token} " \ https://xxxxxxxx x.execute-api.eu-c entral-1.amazonaws .com/Cell_Router_D evelopment/mapper`
	`Orchestrator` Cerca la cella assegnata all'utente e restituisce l'ID della cella e URLs nella seguente risposta:
	`"cellID : cell-0002 , endPoint_1 : https:// xxxxx.execute-api.eu-north-1 .amazonaws.com/ , endPoint_2 : https:// xxxxxxx.execute-api .eu-central-1.amaz onaws.com/"`
		Developer

Eliminazione

Operazione	Description	Competenze richieste
Elimina le risorse.	Per evitare di incorrere in costi aggiuntivi sul tuo account, procedi come segue: 1. Svuota il bucket S3 che hai creato per le funzioni Lambda. 2. Elimina il bucket. 3. Elimina lo stack. CloudFormation	Sviluppatore di app

Risorse correlate

Riferimenti

- Stabilità statica con le zone di disponibilità
- Limiti dell'isolamento dei guasti di AWS: stabilità statica

Video

Physalia: architettura basata su celle per fornire una maggiore disponibilità su Amazon EBS

https://www.youtube-nocookie.com/embed/6 RZMFic Iknq? controlli = 0

Informazioni aggiuntive

Premesse di progettazione dell'architettura basata su celle

Sebbene questo modello si concentri sul router cellulare, è importante comprendere l'intero ambiente. L'ambiente è strutturato in tre livelli distinti:

- Il livello di routing, o Thin layer, che contiene il router cellulare
- Lo strato cellulare, che comprende varie celle
- Il livello Provision and Deploy, che effettua il provisioning delle celle e distribuisce l'applicazione

Ogni livello mantiene la funzionalità anche in caso di alterazioni che interessano altri livelli. Account AWS fungono da limite di isolamento dei guasti.

Il diagramma seguente mostra i livelli ad alto livello. Il livello Cell e il livello Provision and Deploy non rientrano nell'ambito di questo modello.

Per ulteriori informazioni sull'architettura basata su celle, vedere Riduzione dell'ambito di impatto con l'architettura basata su celle: routing delle celle.

Modello di progettazione Cell-router

Il router cellulare è un componente condiviso tra le celle. Per mitigare i potenziali impatti, è importante che il livello di routing utilizzi un design semplicistico e scalabile orizzontalmente che sia il più sottile possibile. Il livello di routing, che funge da punto di ingresso del sistema, è costituito solo dai componenti necessari per assegnare in modo efficiente gli utenti alle celle appropriate. I componenti di questo livello non si occupano della gestione o della creazione di celle.

Questo modello utilizza il routing statico, il che significa che il client memorizza nella cache gli endpoint all'accesso iniziale e successivamente stabilisce una comunicazione diretta con la cella. Vengono avviate interazioni periodiche tra il client e il router cellulare per confermare lo stato corrente o recuperare eventuali aggiornamenti. Questo disaccoppiamento intenzionale consente operazioni ininterrotte per gli utenti esistenti in caso di inattività del router e fornisce funzionalità e resilienza continue all'interno del sistema.

In questo modello, il router cellulare supporta le seguenti funzionalità:

- Recupero dei dati delle celle dal database delle celle nel livello Provision and Deploy e archiviazione o aggiornamento del database locale.
- Assegnazione di una cella a ogni nuovo utente registrato dell'applicazione utilizzando l'algoritmo di assegnazione delle celle.
- Memorizzazione della user-to-cells mappatura nel database locale.
- Verifica della capacità delle celle durante l'assegnazione dell'utente e trasmissione di un evento per il distributore automatico al livello Provision and Deploy per creare celle.
- Utilizzo dell'algoritmo dei criteri di creazione delle celle per fornire questa funzionalità.
- Rispondere alle richieste degli utenti appena registrati fornendo le URLs celle statiche. Queste URLs verranno memorizzate nella cache del client con un time to live (TTL).

- Rispondere alle richieste degli utenti esistenti di un URL non valido fornendo un URL nuovo o aggiornato.

Per comprendere meglio il router cellulare dimostrativo configurato dal CloudFormation modello, esamina i seguenti componenti e passaggi:

1. Configura e configura il pool di utenti di Amazon Cognito.
2. Configura e configura l'API API Gateway per il router cellulare.
3. Crea una tabella DynamoDB.
4. Crea e configura una coda SQS.
5. Implementa il. `Orchestrator`
6. Implementa le funzioni Lambda:`Dispatcher`,,`Scaler`,`Mapper`. `Validator`
7. Valuta e verifica.

Il presupposto è che il livello Provision and Deploy sia già stabilito. I dettagli di implementazione non rientrano nell'ambito di questo artefatto.

Poiché questi componenti sono impostati e configurati da un CloudFormation modello, i passaggi seguenti vengono presentati a un livello descrittivo e di alto livello. Il presupposto è che l'utente disponga delle AWS competenze necessarie per completare l'installazione e la configurazione.

1. Imposta e configura il pool di utenti di Amazon Cognito

Accedi a e apri Console di gestione AWS la console Amazon Cognito all'indirizzo. https://console.aws.amazon.com/cognito/ Configura e configura un pool di utenti Amazon Cognito denominato`CellRouterPool`, con integrazione di app, interfaccia utente ospitata e le autorizzazioni necessarie.

2. Configurare e configurare l'API API Gateway per il router cellulare

Apri la console API Gateway all'indirizzo https://console.aws.amazon.com/apigateway/. Configura e configura un'API denominata`CellRouter`, utilizzando un autorizzatore Amazon Cognito integrato con il pool di utenti di Amazon Cognito. `CellRouterPool` Implementa i seguenti elementi:

- `CellRouter`Risorse API, inclusi `POST` i metodi
- Integrazione con il flusso di lavoro Step Functions implementato nella fase 5
- Autorizzazione tramite l'autorizzatore Amazon Cognito

- Mappature delle richieste di integrazione e delle risposte
- Assegnazione delle autorizzazioni necessarie

3. Creare una tabella DynamoDB

Apri la console DynamoDB https://console.aws.amazon.com/dynamodb/all'indirizzo e crea una tabella DynamoDB standard chiamata con la seguente configurazione: `tbl_router`

- Chiave di partizione – `marketId`
- Chiave di ordinamento – `cellId`
- Modalità di capacità – Fornito
- Point-in-time recovery (PITR) – Disattivato

Nella scheda Indici, crea un indice secondario globale chiamato. `marketId-currentCapacity-index` La funzione `Scaler` Lambda utilizzerà l'indice per condurre ricerche efficienti per la cella con il minor numero di utenti assegnati.

Crea la struttura della tabella con i seguenti attributi:

- `marketId`– Europa
- `cellId`– cell-0002
- `currentCapacity`– 2
- `endPoint_1`– <your endpoint for the first Region>
- `endPoint_2`– <your endpoint for the second Region>
- `IsHealthy`– Vero
- `maxCapacity`– 10
- `regionCode_1` – `eu-north-1`
- `regionCode_2` – `eu-central-1`
- `userIds`– <your email address>

4. Creare e configurare una coda SQS

Apri la console Amazon SQS all'indirizzo https://console.aws.amazon.com/sqs/e crea una coda SQS standard chiamata crittografia a chiave Amazon SQS configurata con **`CellProvisioning`** Amazon SQS.

5. Implementa l'Orchestrator

Sviluppa un flusso di lavoro Step Functions che funga da flusso di lavoro `Orchestrator` per il router. Il flusso di lavoro è richiamabile tramite l'API del router cellulare. Il flusso di lavoro esegue le funzioni Lambda designate in base al percorso della risorsa. Integra la funzione step con l'API Gateway API per il router `CellRouter` cellulare e configura le autorizzazioni necessarie per richiamare le funzioni Lambda.

Il diagramma seguente mostra il flusso di lavoro. Lo stato di scelta richiama una delle funzioni Lambda. Se la funzione Lambda ha esito positivo, il flusso di lavoro termina. Se la funzione Lambda fallisce, viene chiamato lo stato fail.

6. Implementa le funzioni Lambda

Implementa le `Validator` funzioni `Dispatcher` `MapperScaler`,, e. Quando imposti e configuri ciascuna funzione nella dimostrazione, definisci un ruolo per la funzione e assegna le autorizzazioni necessarie per eseguire le operazioni richieste sulla tabella DynamoDB. `tbl_router` Inoltre, integra ogni funzione nel flusso di lavoro. `Orchestrator`

Funzione Dispatcher

La `Dispatcher` funzione è responsabile dell'identificazione e dell'assegnazione di una singola cella statica per ogni nuovo utente registrato. Quando un nuovo utente si registra con l'applicazione globale, la richiesta viene inviata alla `Dispatcher` funzione. La funzione elabora la richiesta utilizzando criteri di valutazione predefiniti come i seguenti:

1. Regione – Seleziona la cella nel mercato in cui si trova l'utente. Ad esempio, se l'utente accede all'applicazione globale dall'Europa, seleziona una cella che utilizza Regioni AWS in Europa.
2. Prossimità o latenza – Seleziona la cella più vicina all'utente Ad esempio, se l'utente accede all'applicazione dall'Olanda, la funzione considera una cella che utilizza Francoforte e l'Irlanda. La decisione relativa alla cella più vicina si basa su metriche come la latenza tra la posizione dell'utente e le regioni della cella. Per questo modello di esempio, le informazioni vengono alimentate staticamente dal livello Provision and Deploy.
3. Health – La `Dispatcher` funzione verifica se la cella selezionata è integra in base allo stato della cella fornito (Healthy = true o false).
4. Capacità – La distribuzione degli utenti si basa sul numero minimo di utenti in una logica cellulare, quindi l'utente viene assegnato alla cella con il minor numero di utenti.

> **Note**
>
> Questi criteri vengono presentati solo per spiegare questo modello di esempio. Per un'implementazione di un router cellulare nella vita reale, è possibile definire criteri più raffinati e utilizzare criteri basati su casi.

`Orchestrator`Richiama la funzione Dispatcher per assegnare gli utenti alle celle. In questa funzione demo, il valore di mercato è un parametro statico definito come. `europe`

La `Dispatcher` funzione valuta se una cella è già assegnata all'utente. Se la cella è già assegnata, la `Dispatcher` funzione restituisce gli estremi della cella. Se all'utente non è assegnata alcuna cella, la funzione cerca la cella con il minor numero di utenti, la assegna all'utente e restituisce gli endpoint. L'efficienza della query di ricerca delle celle è ottimizzata utilizzando l'indice secondario globale.

Funzione Mapper

La `Mapper` funzione sovrintende alla memorizzazione e alla manutenzione delle user-to-cell mappature nel database. Una singola cella viene assegnata a ciascun utente registrato. Ogni cella ne ha due distinte URLs, una per ogni regione AWS. Fungendo da endpoint API ospitati su API Gateway, URLs funzionano come punti in entrata verso l'applicazione globale.

Quando la `Mapper` funzione riceve una richiesta dall'applicazione client, esegue una query sulla `tbl_router` tabella DynamoDB per recuperare user-to-cell la mappatura associata all'ID e-mail fornito. Se trova una cella assegnata, la `Mapper` funzione fornisce prontamente le due della cella. URLs La `Mapper` funzione monitora inoltre attivamente le modifiche alla cella URLs e avvia notifiche o aggiornamenti alle impostazioni dell'utente.

Funzione Scaler

La `Scaler` funzione gestisce la capacità residua della cella. Per ogni nuova richiesta di registrazione utente, la `Scaler` funzione valuta la capacità disponibile della cella assegnata dalla `Dispatcher` funzione all'utente. Se la cella ha raggiunto il limite predeterminato in base ai criteri di valutazione specificati, la funzione avvia una richiesta tramite una coda Amazon SQS al livello Provision and Deploy, richiedendo il provisioning e l'implementazione di nuove celle. Il ridimensionamento delle celle può essere eseguito in base a una serie di criteri di valutazione come i seguenti:

1. Numero massimo di utenti – Ogni cella può avere un numero massimo di utenti 500.

2. Capacità buffer – La capacità buffer di ogni cella è del 20 percento, il che significa che ogni cella può essere assegnata a 400 utenti in qualsiasi momento. Il restante 20% della capacità buffer è riservato a casi d'uso futuri e alla gestione di scenari imprevisti (ad esempio, quando i servizi di creazione e provisioning delle celle non sono disponibili).
3. Creazione di celle – Non appena una cella esistente raggiunge il 70 percento della capacità, viene attivata una richiesta per creare una cella aggiuntiva.

> **Note**
>
> Questi criteri vengono presentati solo per spiegare questo modello di esempio. Per un'implementazione di un router cellulare nella vita reale, è possibile definire criteri più raffinati e utilizzare criteri basati su casi.

Il `Scaler` codice dimostrativo viene eseguito da `Orchestrator` after che assegna `Dispatcher` correttamente una cella all'utente appena registrato. Al ricevimento dell'ID della cella da`Dispatcher`, valuta se la cella designata ha una capacità adeguata per ospitare utenti aggiuntivi, sulla base di criteri di valutazione predefiniti. `Scaler` Se la capacità della cella è insufficiente, la `Scaler` funzione invia un messaggio al servizio Amazon SQS. Questo messaggio viene recuperato dal servizio all'interno del livello Provision and Deploy, avviando il provisioning di una nuova cella.

Funzione di validazione

La `Validator` funzione identifica e risolve i problemi relativi all'accesso alle celle. Quando un utente accede all'applicazione globale, l'applicazione recupera le celle URLs dalle impostazioni del profilo utente e indirizza le richieste degli utenti a una delle due regioni assegnate all'interno della cella. Se URLs sono inaccessibili, l'applicazione può inviare una richiesta URL di convalida al router cellulare. Il `Orchestrator` cell-router richiama il. `Validator` `Validator`Avvia il processo di convalida. La convalida può includere, tra gli altri controlli, quanto segue:

- Cella con riferimenti incrociati URLs nella richiesta e URLs memorizzata nel database per identificare ed elaborare potenziali aggiornamenti
- Esecuzione di un controllo approfondito dello stato (ad esempio, una `HTTP GET` richiesta all'endpoint della cella)

In conclusione, la `Validator` funzione fornisce risposte alle richieste delle applicazioni dei clienti, fornendo lo stato di convalida e le eventuali misure correttive necessarie.

`Validator`È progettato per migliorare l'esperienza dell'utente. Prendiamo in considerazione uno scenario in cui alcuni utenti incontrano difficoltà ad accedere all'applicazione globale perché un incidente causa la temporanea indisponibilità delle celle. Invece di presentare errori generici, la `Validator` funzione può fornire istruzioni per la correzione. Questi passaggi potrebbero includere le seguenti azioni:

- Informare gli utenti sull'incidente.
- Fornisci un tempo di attesa approssimativo prima della disponibilità del servizio.
- Fornisci un numero di contatto dell'assistenza per ottenere ulteriori informazioni.

Il codice demo della `Validator` funzione verifica che la cella fornita dall'utente URLs nella richiesta corrisponda ai record memorizzati nella `tbl_router` tabella. La `Validator` funzione verifica anche se le cellule sono sane.

Configura l'accesso privato a un bucket Amazon S3 tramite un endpoint VPC

Martin Maritsch, Nicolas Jacob Baer, Gabriel Rodriguez Garcia, Shukhrat Khodjaev, Mohan Gowda Purushothama e Joaquin Rinaudo, Amazon Web Services

Riepilogo

In Amazon Simple Storage Service (Amazon S3), le impostazioni URLs predefinite consentono di condividere file di dimensioni arbitrarie con utenti target. Per impostazione predefinita, le versioni predefinite di Amazon S3 URLs sono accessibili da Internet entro una finestra temporale di scadenza, il che le rende comode da usare. Tuttavia, gli ambienti aziendali spesso richiedono l'accesso ad Amazon S3 preimpostato URLs per essere limitato a una sola rete privata.

Questo modello presenta una soluzione serverless per interagire in modo sicuro con gli oggetti S3 utilizzando dati predefiniti da una rete privata senza attraversamento di Internet URLs . Nell'architettura, gli utenti accedono a un Application Load Balancer tramite un nome di dominio interno. Il traffico viene instradato internamente tramite Amazon API Gateway e un endpoint di cloud privato virtuale (VPC) per il bucket S3. La AWS Lambda funzione genera predefiniti URLs per il download di file tramite l'endpoint VPC privato, che aiuta a migliorare la sicurezza e la privacy dei dati sensibili.

Prerequisiti e limitazioni

Prerequisiti

- Un VPC che include una sottorete distribuita in e connessa alla rete aziendale (ad esempio, tramite). Account AWS AWS Direct Connect

Limitazioni

- Il bucket S3 deve avere lo stesso nome del dominio, quindi ti consigliamo di controllare le regole di denominazione dei bucket Amazon S3.
- Questa architettura di esempio non include funzionalità di monitoraggio per l'infrastruttura distribuita. Se il tuo caso d'uso richiede il monitoraggio, prendi in considerazione l'aggiunta di servizi AWS di monitoraggio.

- Questa architettura di esempio non include la convalida dell'input. Se il tuo caso d'uso richiede la convalida degli input e un maggiore livello di sicurezza, prendi in considerazione l'utilizzo AWS WAF per proteggere la tua API.
- Questa architettura di esempio non include la registrazione degli accessi con Application Load Balancer. Se il tuo caso d'uso richiede la registrazione degli accessi, valuta la possibilità di abilitare i log di accesso del Load Balancer.

Versioni

- Python versione 3.11 o successiva
- Terraform versione 1.6 o successiva

Architecture

Stack tecnologico Target

I seguenti servizi AWS vengono utilizzati nello stack tecnologico di destinazione:

- Amazon S3 è il servizio di storage principale utilizzato per caricare, scaricare e archiviare file in modo sicuro.
- Amazon API Gateway espone risorse ed endpoint per l'interazione con il bucket S3. Questo servizio svolge un ruolo nella generazione di dati predefiniti per il download o il caricamento. URLs
- AWS Lambdagenera predefiniti URLs per il download di file da Amazon S3. La funzione Lambda viene chiamata da API Gateway.
- Amazon VPC distribuisce risorse all'interno di un VPC per garantire l'isolamento della rete. Il VPC include sottoreti e tabelle di routing per controllare il flusso di traffico.
- Application Load Balancer indirizza il traffico in entrata verso API Gateway o verso l'endpoint VPC del bucket S3. Consente agli utenti della rete aziendale di accedere alle risorse internamente.
- L'endpoint VPC per Amazon S3 consente la comunicazione diretta e privata tra le risorse nel VPC e Amazon S3 senza attraversare la rete Internet pubblica.
- AWS Identity and Access Management (IAM) controlla l'accesso alle risorse. AWS Le autorizzazioni sono impostate per garantire interazioni sicure con l'API e altri servizi.

Architettura Target

Il diagramma illustra quanto segue:

1. Gli utenti della rete aziendale possono accedere all'Application Load Balancer tramite un nome di dominio interno. Partiamo dal presupposto che esista una connessione tra la rete aziendale e la sottorete intranet in Account AWS (ad esempio, tramite una connessione). Direct Connect
2. L'Application Load Balancer indirizza il traffico in entrata verso API Gateway per generare dati predefiniti URLs per il download o il caricamento su Amazon S3 o verso l'endpoint VPC del bucket S3. In entrambi gli scenari, le richieste vengono instradate internamente e non devono attraversare Internet.
3. API Gateway espone risorse ed endpoint per interagire con il bucket S3. In questo esempio, forniamo un endpoint per scaricare file dal bucket S3, ma questo potrebbe essere esteso per fornire anche funzionalità di caricamento.
4. La funzione Lambda genera l'URL predefinito per scaricare un file da Amazon S3 utilizzando il nome di dominio dell'Application Load Balancer anziché il dominio pubblico Amazon S3.
5. L'utente riceve l'URL predefinito e lo utilizza per scaricare il file da Amazon S3 utilizzando Application Load Balancer. Il load balancer include un percorso predefinito per inviare il traffico non destinato all'API verso l'endpoint VPC del bucket S3.
6. L'endpoint VPC indirizza l'URL predefinito con il nome di dominio personalizzato al bucket S3. Il bucket S3 deve avere lo stesso nome del dominio.

Automazione e scalabilità

Questo modello utilizza Terraform per distribuire l'infrastruttura dal repository di codice in un. Account AWS

Tools (Strumenti)

Strumenti

- Python è un linguaggio di programmazione per computer generico.
- Terraform è uno strumento Infrastructure as Code (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire con i AWS servizi tramite comandi nella shell della riga di comando.

Archivio di codice

Il codice per questo pattern è disponibile in un GitHub repository all'indirizzo https://github.com/aws-samples/ private-s3-vpce.

Best practice

L'architettura di esempio per questo pattern utilizza le autorizzazioni IAM per controllare l'accesso all'API. Chiunque disponga di credenziali IAM valide può chiamare l'API. Se il tuo caso d'uso richiede un modello di autorizzazione più complesso, potresti voler utilizzare un meccanismo di controllo degli accessi diverso.

Epiche

Implementa la soluzione in un Account AWS

Operazione	Description	Competenze richieste
Ottenere AWS le credenziali.	Controlla AWS le tue credenziali e il tuo accesso al tuo account. Per istruzioni, consulta Impostazioni dei file di configurazione e credenziali nella AWS CLI documentazione.	AWS DevOps, Informazioni generali su AWS
Clonare il repository.	Clona il GitHub repository fornito con questo modello: `git clone https://github.com/aws-samples/private-s3-vpce`	AWS DevOps, Informazioni generali su AWS
Configura le variabili.	1. Sul tuo computer, nel GitHub repository, apri la `terraform` cartella: `cd terraform` 2. Apri il `example.tfvars` file e personalizza i	AWS DevOps, Informazioni generali su AWS

Operazione	Description	Competenze richieste
	parametri in base alle tue esigenze.	
Implementa una soluzione.	1. Nella `terraform` cartella, esegui Terraform e passa le variabili che hai personalizzato: `terraform apply -var-file="example.tfvars"` 2. Verifica che le risorse mostrate nel diagramma di architettura siano state distribuite correttamente.	AWS DevOps, Informazioni generali su AWS

Test della soluzione

Operazione	Description	Competenze richieste
Crea un file di test.	Carica un file su Amazon S3 per creare uno scenario di test per il download del file. Puoi utilizzare la console Amazon S3 o il seguente AWS CLI comando: `aws s3 cp /path/to/testfile s3://your-bucket-name/testfile`	AWS DevOps, Informazioni generali su AWS
Prova la funzionalità degli URL predefiniti.	1. Invia una richiesta all'Application Load Balancer per creare un URL predefinito	AWS DevOps, Informazioni generali su AWS

Operazione	Description	Competenze richieste
	per il file di test utilizzando awscurl: `awscurl https://your-domain-name/api/get_url?key=testfile` Questo passaggio crea una firma valida dalle tue credenziali, che verrà convalidata da API Gateway. 2. Analizza il link contenuto nella risposta ricevuta nel passaggio precedente e apri l'URL predefinito per scaricare il file.	
Elimina.	Assicurati di rimuovere le risorse quando non sono più necessarie: `terraform destroy`	AWS DevOps, Informazioni generali su AWS

risoluzione dei problemi

Problema	Soluzione
I nomi delle chiavi degli oggetti S3 con caratteri speciali come i segni numerici (#) interrompono i parametri URL e generano errori.	Codifica correttamente i parametri URL e assicurati che il nome della chiave dell'oggetto S3 segua le linee guida di Amazon S3.

Risorse correlate

Amazon S3:

- Condivisione di oggetti con predefiniti URLs
- Controllo dell'accesso dagli endpoint VPC con policy bucket

Amazon API Gateway:

- Usa le policy degli endpoint VPC per uso privato APIs in API Gateway

Application Load Balancer:

- Hosting di siti Web statici HTTPS interni con ALB, S3 e PrivateLink (AWS post del blog)

Risolvi i problemi relativi agli stati AWS Step Functions utilizzando Amazon Bedrock

Aniket Kurzadkar e Sangam Kushwaha, Amazon Web Services

Riepilogo

AWS Step Functions le funzionalità di gestione degli errori possono aiutarti a individuare un errore che si verifica durante uno stato di un flusso di lavoro, ma può comunque essere difficile individuare la causa principale di un errore ed eseguirne il debug. Questo modello affronta questa sfida e mostra come Amazon Bedrock può aiutarti a risolvere gli errori che si verificano durante gli stati di Step Functions.

Step Functions fornisce l'orchestrazione del flusso di lavoro, semplificando l'automazione dei processi per gli sviluppatori. Step Functions offre anche funzionalità di gestione degli errori che offrono i seguenti vantaggi:

- Gli sviluppatori possono creare applicazioni più resilienti che non falliscono completamente quando qualcosa va storto.
- I flussi di lavoro possono includere la logica condizionale per gestire diversi tipi di errori in modo diverso.
- Il sistema può riprovare automaticamente le operazioni non riuscite, magari con un backoff esponenziale.
- È possibile definire percorsi di esecuzione alternativi per gli scenari di errore, consentendo al flusso di lavoro di adattarsi e continuare l'elaborazione.

Quando si verifica un errore in un flusso di lavoro Step Functions, questo modello mostra come inviare il messaggio di errore e il contesto a un modello base (FM) come Claude 3 supportato da Step Functions. L'FM può analizzare l'errore, classificarlo e suggerire potenziali passaggi di correzione.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Conoscenza di base AWS Step Functions e flussi di lavoro
- Connettività dell'API Amazon Bedrock

Limitazioni

- Puoi utilizzare l'approccio di questo modello per vari Servizi AWS scopi. Tuttavia, i risultati potrebbero variare in base al prompt creato e successivamente valutato da AWS Lambda Amazon Bedrock.
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità regionale, consulta i servizi AWS per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Architecture

Il diagramma seguente mostra i componenti del flusso di lavoro e dell'architettura per questo modello.

Il diagramma mostra il flusso di lavoro automatizzato per la gestione e la notifica degli errori in una macchina a stati Step Functions:

1. Lo sviluppatore avvia l'esecuzione di una macchina a stati.
2. La macchina a stati Step Functions inizia a elaborare i suoi stati. I risultati possibili sono due:
   - (a) Se tutti gli stati vengono eseguiti correttamente, il flusso di lavoro passa direttamente ad Amazon SNS per una notifica di successo via e-mail.
   - (b) Se uno stato fallisce, il flusso di lavoro passa alla funzione Lambda per la gestione degli errori.
3. In caso di errore, si verifica quanto segue:
   - (a) Viene attivata la funzione Lambda (gestore degli errori). La funzione Lambda estrae il messaggio di errore dai dati dell'evento che la macchina a stati Step Functions le ha passato. Quindi la funzione Lambda prepara un prompt basato su questo messaggio di errore e lo invia ad Amazon Bedrock. Il prompt richiede soluzioni e suggerimenti relativi all'errore specifico riscontrato.
   - (b) Amazon Bedrock, che ospita il modello di intelligenza artificiale generativa, elabora la richiesta di input. (Questo modello utilizza il modello di base Anthropic Claude 3 (FM), uno dei tanti supportati da FMs Amazon Bedrock.) Il modello di intelligenza artificiale analizza il contesto dell'errore. Quindi il modello genera una risposta che può includere spiegazioni del motivo per cui si è verificato l'errore, potenziali soluzioni per risolverlo e suggerimenti per evitare di ripetere gli stessi errori in futuro.

Amazon Bedrock restituisce la risposta generata dall'intelligenza artificiale alla funzione Lambda. La funzione Lambda elabora la risposta, potenzialmente formattandola o estraendo informazioni chiave. Quindi la funzione Lambda invia la risposta all'output della macchina a stati.

4. Dopo la gestione degli errori o l'esecuzione corretta, il flusso di lavoro si conclude attivando Amazon SNS per l'invio di una notifica e-mail.

Tools (Strumenti)

Servizi AWS

- Amazon Bedrock è un servizio completamente gestito che rende disponibili per l'uso modelli di base ad alte prestazioni (FMs) delle principali startup di intelligenza artificiale e di Amazon tramite un'API unificata.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.
- AWS Step Functionsè un servizio di orchestrazione senza server che ti aiuta a combinare AWS Lambda funzioni e altro per creare applicazioni aziendali critiche. Servizi AWS

Best practice

- Poiché Amazon Bedrock è un modello di intelligenza artificiale generativo che apprende dai dati addestrati, utilizza tali dati anche per addestrare e generare contesto. Come best practice, nascondi tutte le informazioni private che potrebbero causare problemi di fuga di dati.
- Sebbene l'intelligenza artificiale generativa possa fornire informazioni preziose, le decisioni critiche relative alla gestione degli errori dovrebbero comunque coinvolgere la supervisione umana, specialmente negli ambienti di produzione.

Epiche

Crea una macchina a stati per il tuo flusso di lavoro

Operazione	Description	Competenze richieste
Crea una macchina a stati.	Per creare una macchina a stati adatta al tuo flusso di lavoro, procedi come segue: 1. Accedi a e apri la AWS Step Functions console. Console di gestione AWS 2. Dal riquadro di navigazione a sinistra, scegli Macchine a stati. 3. Scegli Create state machine (Crea macchina a stati). 4. Scegli un modello in base al tuo caso d'uso o scegli Blank per creare un modello in base alle tue esigenze.	AWS DevOps

Creazione di una funzione Lambda

Operazione	Description	Competenze richieste
Creazione di una funzione Lambda.	Per creare una funzione Lambda, procedi come segue: 1. In Console di gestione AWS, accedi alla AWS Lambda console. 2. Nel riquadro di navigazione a sinistra, scegli Funzioni, quindi scegli Crea funzione.	AWS DevOps

Operazione	Description	Competenze richieste
	3. Nella pagina Crea funzione, scegli tra le opzioni per creare una funzione. Quindi, inserisci un nome in Nome della funzione e scegli la lingua appropria ta dall'elenco a discesa in Runtime. 4. Scegli Crea funzione.	

Operazione	Description	Competenze richieste

Imposta la logica richiesta nel codice Lambda.	• Per connetterti all'API Amazon Bedrock utilizzando il AWS SDK per Python (Boto3), usa il codice seguente.
	
	Questo codice configura un client per Amazon Bedrock, prepara i parametri necessari e quindi invia una richiesta al modello Claude 3 con un prompt specificato.
	
	Questo pattern richiama il modello Claude 3. Per ulteriori informazioni su tutti i modelli di base supportati, incluso il modello correlato IDs, consulta Modelli di base supportati in Amazon Bedrock nella documentazione di Amazon Bedrock.
	
	`client = boto3.client( service_name="bedrock-runtime", region_name="selected-region" )`
	
	`# Invoke Claude 3 with the text prompt model_id = "your-model-id" # Select your Model ID, Based on the Model Id, Change the body format`
		AWS DevOps

Operazione	Description	Competenze richieste

	<pre>try:
	response = client.invoke_model(
	modelId=model_id,
	body=json.dumps(
	{
	"anthropic_version": "bedrock-2023-05-31",
	"max_tokens": 1024,
	"messages": [
	{
	"role": "user",
	"content": [{"type": "text", "text": prompt}],
	}
	],
	}
	),
	)</pre> • (Facoltativo) Sostituisci l'account Account AWS IDs con un segnaposto. IDs Per motivi di sicurezza, questo approccio può essere utile per ripulire registri, messaggi di errore o altri output che potrebber	

Operazione	Description	Competenze richieste

	o contenere informazioni riservate sull'account.
	
	Il codice seguente troverà qualsiasi occorrenza di un numero di 12 cifre racchiuso tra due punti (che è il formato di Account AWS IDs Amazon Resource Names (ARNs) e alcuni altri AWS identificatori) e lo sostituirà con l'ID dell'account segnaposto. `":123456789012:"`
	
	`def replace_account_id(input_string):`
	`  # Use a regular expression to find the AWS account ID pattern`
	`  account_id_pattern = r'(:\d{12}:)'`
	`  # Replace the matched pattern with ":123456789012:"`
	`  modified_string = re.sub(account_id_pattern, ":123456789012:", input_string)`
	`  return modified_string`	

Integrazione di Step Functions con Lambda

Operazione	Description	Competenze richieste
Configura Lambda per gestire gli errori in Step Functions.	Per configurare Step Functions in modo da gestire gli errori senza interrompere il flusso di lavoro, procedi come segue: 1. Nella console Step Functions, accedi alla macchina a stati creata in precedenza. 2. Scegliete Modifica, quindi scegliete il servizio per cui desiderate impostare la gestione degli errori e scegliete Gestione degli errori. 3. Scegli Aggiungi nuovo catcher e, per lo stato di fallback, scegli Lambda, quindi scegli la funzione Lambda che hai creato in precedenza. Per ulteriori informazioni, consulta Catch errors nella documentazione di Step Functions.	AWS DevOps

Risoluzione dei problemi

Problema	Soluzione
Lambda non può accedere all'API Amazon Bedrock (non autorizzata a eseguire)	Questo errore si verifica quando il ruolo Lambda non dispone dell'autorizzazion e per accedere all'API Amazon Bedrock. Per risolvere questo problema, aggiungi la `AmazonBedrockFullAccess` policy per il ruolo Lambda. Per ulteriori informazioni, consulta AmazonBedrockFullAccessla AWS Managed Policy Reference Guide.
Errore di timeout Lambda	A volte possono essere necessari più di 30 secondi per generare una risposta e inviarla indietro, a seconda della richiesta. Per risolvere questo problema, aumenta il tempo di configura zione. Per ulteriori informazioni, consulta Configura il timeout della funzione Lambda nella Guida per gli AWS Lambda sviluppatori.

Risorse correlate

- Amazon Bedrock
- Accesso all'API Amazon Bedrock
- Crea la tua prima funzione Lambda
- Sviluppo di flussi di lavoro con Step Functions
- AWS Step Functions

Altri modelli

- Accedi, esegui query e unisciti a tabelle Amazon DynamoDB utilizzando Athena
- Automatizza le CodeGuru revisioni Amazon per le applicazioni AWS CDK Python utilizzando Actions GitHub
- Automatizza la valutazione delle risorse AWS
- Automatizza la distribuzione di applicazioni annidate utilizzando AWS SAM
- Automatizza l'implementazione dei Catena di approvvigionamento di AWS data lake in una configurazione multi-repository utilizzando GitHub Actions, Artifactory e Terraform
- Automatizza la replica delle istanze Amazon RDS su Account AWS
- Archivia automaticamente gli elementi su Amazon S3 utilizzando DynamoDB TTL
- Rileva automaticamente le modifiche e avvia diverse CodePipeline pipeline per un monorepo in CodeCommit
- Crea un'architettura serverless multi-tenant in Amazon Service OpenSearch
- Crea un visualizzatore di file mainframe avanzato nel cloud AWS
- Calcola il valore a rischio (VaR) utilizzando i servizi AWS
- Copia i prodotti AWS Service Catalog su diversi account AWS e regioni AWS
- Crea automaticamente pipeline CI dinamiche per progetti Java e Python
- Scomponi i monoliti in microservizi utilizzando CQRS e l'event sourcing
- Implementa un'applicazione a pagina singola basata su React su Amazon S3 e CloudFront
- Implementa un'API Amazon API Gateway su un sito Web interno utilizzando endpoint privati e un Application Load Balancer
- Implementa e gestisci un data lake serverless sul cloud AWS utilizzando l'infrastruttura come codice
- Implementa uno use case RAG AWS utilizzando Terraform e Amazon Bedrock
- Sviluppa un assistente basato su chat completamente automatizzato utilizzando gli agenti e le knowledge base di Amazon Bedrock
- Sviluppa assistenti avanzati basati sull'intelligenza artificiale generativa utilizzando RAG e suggerimenti ReAct
- Genera dinamicamente una policy IAM con IAM Access Analyzer utilizzando Step Functions
- Incorpora i componenti visivi di Amazon Quick Sight nelle applicazioni Web utilizzando Amazon Cognito e l'automazione IAc

- Assicurati che la registrazione di Amazon EMR su Amazon S3 sia abilitata al momento del lancio
- Stima del costo di una tabella DynamoDB per la capacità su richiesta
- Genera consigli personalizzati e riclassificati con Amazon Personalize
- Genera dati di test utilizzando un job AWS Glue e Python
- SHA1 Implementa l'hashing per i dati PII durante la migrazione da SQL Server a PostgreSQL
- Implementa il modello di saga serverless utilizzando AWS Step Functions
- Migliora le prestazioni operative abilitando Amazon DevOps Guru su più regioni AWS, account e OUs con AWS CDK
- Avvia un CodeBuild progetto su più account AWS utilizzando Step Functions e una funzione proxy Lambda
- Esegui la migrazione dei carichi di lavoro Apache Cassandra su Amazon Keyspaces utilizzando AWS Glue
- Monitora l'uso di un'Amazon Machine Image condivisa su più Account AWS
- Ottimizza le implementazioni serverless multi-account utilizzando i flussi di lavoro e Actions AWS CDK GitHub
- Orchestra una pipeline ETL con convalida, trasformazione e partizionamento utilizzando AWS Step Functions
- Interroga le tabelle Amazon DynamoDB con SQL utilizzando Amazon Athena
- Esegui carichi di lavoro pianificati e basati su eventi su larga scala con AWS Fargate
- Invia attributi personalizzati ad Amazon Cognito e inseriscili in token
- Distribuisci contenuti statici in un bucket Amazon S3 tramite un VPC utilizzando Amazon CloudFront
- Semplifica lo sviluppo e la distribuzione di bot Amazon Lex utilizzando un flusso di lavoro automatizzato
- Struttura un progetto Python in architettura esagonale utilizzando AWS Lambda
- Traduci il linguaggio naturale in query DSL for OpenSearch ed Elasticsearch
- Scarica i dati da un cluster Amazon Redshift tra più account su Amazon S3
- Coordina la dipendenza dalle risorse e l'esecuzione delle attività utilizzando il costrutto AWS Fargate WaitCondition hook
- Usa gli agenti Amazon Bedrock per automatizzare la creazione di controlli di accesso in Amazon EKS tramite istruzioni basate su testo

Rete

Argomenti

- Automatizza la configurazione del peering interregionale con AWS Transit Gateway
- Centralizza la connettività di rete utilizzando AWS Transit Gateway
- Configurare la crittografia HTTPS per Oracle JD Edwards EnterpriseOne su Oracle WebLogic utilizzando un Application Load Balancer
- Connect ai dati e ai piani di controllo dell'Application Migration Service tramite una rete privata
- Crea oggetti Infoblox utilizzando risorse CloudFormation personalizzate AWS e Amazon SNS
- Crea un'architettura IPAM gerarchica e multiregionale utilizzando Terraform AWS
- Personalizza CloudWatch gli avvisi Amazon per AWS Network Firewall
- Distribuisci risorse in una AWS Wavelength zona utilizzando Terraform
- Esegui la migrazione di record DNS in blocco verso una zona ospitata privata di Amazon Route 53
- Modifica le intestazioni HTTP durante la migrazione da F5 a un Application Load Balancer su AWS
- Crea un rapporto sui risultati di Network Access Analyzer per l'accesso a Internet in entrata in più Account AWS
- Configura la risoluzione DNS per reti ibride in un ambiente multi-account AWS
- Verificare che i sistemi di bilanciamento del carico ELB richiedano la terminazione TLS
- Visualizza i log e i parametri di AWS Network Firewall utilizzando Splunk
- Altri modelli

Automatizza la configurazione del peering interregionale con AWS Transit Gateway

Ram Kandaswamy, Amazon Web Services

Riepilogo

AWS Transit Gateway collega cloud privati virtuali (VPCs) e reti locali tramite un hub centrale. Il traffico Transit Gateway rimane sempre sulla dorsale globale di Amazon Web Services (AWS) e non attraversa la rete Internet pubblica, il che riduce i vettori di minaccia, come gli exploit comuni e gli attacchi Distributed Denial of Service (S). DDo

Se devi comunicare tra due o più regioni AWS, puoi utilizzare il peering di Transit Gateway inter-regionale per stabilire connessioni peering tra gateway di transito in diverse regioni. Tuttavia, la configurazione manuale del peering interregionale con Transit Gateway può essere un processo che richiede molto tempo e prevede più passaggi. Questo modello fornisce un processo automatizzato per rimuovere questi passaggi manuali utilizzando il codice per eseguire il peering. Puoi utilizzare questo approccio se devi configurare ripetutamente diverse regioni e account AWS durante una configurazione di un'organizzazione multiregionale.

Questo modello utilizza uno CloudFormation stack AWS che include il flusso di lavoro AWS Step Functions, le funzioni AWS Lambda, i ruoli AWS Identity and Access Management (IAM) e i gruppi di log in Amazon CloudWatch Logs. È quindi possibile avviare un'esecuzione di Step Functions e creare la connessione peering interregionale per i gateway di transito. Per configurare manualmente il peering interregionale, consulta Peer in VPCs diverse regioni AWS utilizzando AWS Transit Gateway.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un bucket Amazon Simple Storage Service (Amazon S3) esistente.
- Gateway di transito, creati e configurati nella regione richiedente e nelle regioni accettanti. La regione richiedente è il luogo in cui viene originata una richiesta di peering e le regioni accettanti accettano la richiesta di peering. Per ulteriori informazioni su questo argomento, consulta Creazione e accettazione di una connessione peering VPC nella documentazione di Amazon VPC.

- VPCs, installato e configurato nelle regioni accettante e richiedente. Per i passaggi per creare un VPC, consulta Creare il VPC da Get Started with Amazon VPC nella documentazione di Amazon VPC.
- È VPCs necessario utilizzare il tag e il valore. `addToTransitGateway` `true`
- Gruppi di sicurezza e liste di controllo degli accessi alla rete (ACLs) per te VPCs, configurati in base alle tue esigenze. Per ulteriori informazioni su questo argomento, consulta Gruppi di sicurezza per il tuo VPC e la tua rete ACLs nella documentazione di Amazon VPC.

Regioni e limitazioni AWS

- Solo alcune regioni AWS supportano il peering interregionale. Per un elenco completo delle regioni che supportano il peering interregionale, consulta AWS Transit Gateway. FAQs
- Nel codice di esempio allegato, si presume che la regione richiedente sia e si presume che lo sia `us-east-2` la regione accettante. `us-west-2` Se vuoi configurare regioni diverse, devi modificare questi valori in tutti i file Python. Per implementare una configurazione più complessa che coinvolga più di due regioni, puoi modificare la Step Function per passare le Regions come parametro alla funzione Lambda ed eseguire la funzione per ogni combinazione.

Architecture

Il diagramma mostra un flusso di lavoro con i seguenti passaggi:

1. L'utente crea uno CloudFormation stack AWS.
2. AWS CloudFormation crea una macchina a stati Step Functions che utilizza una funzione Lambda. Per ulteriori informazioni su questo argomento, consulta Creazione di una macchina a stati Step Functions che utilizza Lambda nella documentazione di AWS Step Functions.
3. Step Functions richiama una funzione Lambda per il peering.
4. La funzione Lambda crea una connessione peering tra i gateway di transito.
5. Step Functions richiama una funzione Lambda per le modifiche alla tabella delle rotte.
6. La funzione Lambda modifica le tabelle delle rotte aggiungendo il blocco Classless Inter-Domain Routing (CIDR) di. VPCs

Flusso di lavoro Step Functions

Il diagramma mostra il seguente flusso di lavoro Step Functions:

1. Il flusso di lavoro Step Functions richiama la funzione Lambda per il peering del gateway di transito.
2. È prevista una chiamata con timer di attesa di un minuto.
3. Lo stato di peering viene recuperato e inviato al blocco delle condizioni. Il blocco è responsabile del looping.
4. Se la condizione di successo non viene soddisfatta, il flusso di lavoro viene codificato per entrare nella fase del timer.
5. Se viene soddisfatta la condizione di successo, viene chiamata una funzione Lambda per modificare le tabelle delle rotte. Dopo questa chiamata, il flusso di lavoro Step Functions termina.

Tools (Strumenti)

- AWS CloudFormation: AWS CloudFormation è un servizio che ti aiuta a modellare e configurare le tue risorse AWS.
- Amazon CloudWatch Logs — CloudWatch Logs ti aiuta a centralizzare i log di tutti i tuoi sistemi, applicazioni e servizi AWS che utilizzi.
- AWS Identity and Access Management (IAM): IAM è un servizio Web per controllare in modo sicuro l'accesso ai servizi AWS.
- AWS Lambda: Lambda esegue il codice su un'infrastruttura di calcolo ad alta disponibilità ed esegue tutta l'amministrazione delle risorse di calcolo.
- AWS Step Functions — Step Functions semplifica il coordinamento dei componenti delle applicazioni distribuite come una serie di passaggi in un flusso di lavoro visivo.

Epiche

Automatizza il peering

Operazione	Description	Competenze richieste
Carica i file allegati nel tuo bucket S3.	Accedi alla Console di gestione AWS, apri la console	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	Amazon S3, quindi carica `get-transit-gateway-peering-status.zip` i file e i file (allegati) nel `modify-transit-gateway-routes.zip` tuo bucket S3. `peer-transit-gateway.zip`	

Operazione	Description	Competenze richieste
Crea lo CloudFormation stack AWS.	Esegui il seguente comando per creare uno CloudFormation stack AWS utilizzando il `transit-gateway-peering.json` file (allegato): `aws cloudformation create-stack --stack-name myteststack --template-body file://sampletemplate.json` Lo CloudFormation stack AWS crea il flusso di lavoro Step Functions, le funzioni Lambda, i ruoli IAM CloudWatch e i gruppi di log. Assicurati che il CloudFormation modello AWS si riferisca al bucket S3 che contiene i file che hai caricato in precedenza. **Note** Puoi anche creare uno stack utilizzando la CloudFormation console AWS. Per ulteriori informazioni su questo argomento, consulta Creazione di uno stack sulla CloudFormation console AWS nella	DevOps ingegnere

Operazione	Description	Competenze richieste
	CloudFormation documentazione AWS.	
Inizia una nuova esecuzione in Step Functions.	Apri la console Step Functions e avvia una nuova esecuzione. Step Functions richiama la funzione Lambda e crea la connessione peering per i gateway di transito. Non è necessario un file JSON di input. Verifica che sia disponibile un allegato e che il tipo di connessione sia Peering. Per ulteriori informazioni su questo argomento, consulta Start a new execution from Getting started with AWS Step Functions nella documentazione di AWS Steps Functions.	DevOps ingegnere, General AWS

Operazione	Description	Competenze richieste
Verifica i percorsi nelle tabelle dei percorsi.	Il peering interregionale viene stabilito tra i gateway di transito. Le tabelle delle rotte vengono aggiornate con l'intervallo di blocchi CIDR della regione peer VPC. IPv4 Apri la console Amazon VPC e scegli la scheda Associazioni nella tabella delle rotte che corrisponde all'allegato del gateway di transito. Verifica l'intervallo di blocchi VPC CIDR delle regioni peerizzate. Per passaggi e istruzioni dettagliati, consulta Associare una tabella di routing del gateway di transito nella documentazione di Amazon VPC.	Amministratore di rete

Risorse correlate

- Esecuzioni in Step Functions
- Allegati di peering Transit Gateway
- Effettua il VPCs peering in diverse regioni AWS utilizzando AWS Transit Gateway
- Interconnessione VPCs tra regioni AWS tramite AWS Transit Gateway - Demo (video)

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Centralizza la connettività di rete utilizzando AWS Transit Gateway

Mydhili Palagummi e Nikhil Marrapu, Amazon Web Services

Riepilogo

Questo modello descrive la configurazione più semplice in cui AWS Transit Gateway può essere utilizzato per connettere una rete locale a cloud privati virtuali (VPCs) in più account AWS all'interno di una regione AWS. Utilizzando questa configurazione, puoi stabilire una rete ibrida che collega più reti VPC in una regione e una rete locale. Ciò si ottiene utilizzando un gateway di transito e una connessione di rete privata virtuale (VPN) alla rete locale.

Prerequisiti e limitazioni

Prerequisiti

- Un account per i servizi di rete di hosting, gestito come account membro di un'organizzazione in AWS Organizations
- VPCs in più account AWS, senza sovrapposizione di blocchi Classless Inter-Domain Routing (CIDR)

Limitazioni

Questo modello non supporta l'isolamento del traffico tra determinate reti o tra quelle locali. VPCs Tutte le reti collegate al gateway di transito saranno in grado di raggiungersi. Per isolare il traffico, è necessario utilizzare tabelle di routing personalizzate sul gateway di transito. Questo modello collega solo la rete locale VPCs e quella locale utilizzando un'unica tabella di routing del gateway di transito predefinita, che è la configurazione più semplice.

Architecture

Stack tecnologico Target

- AWS Transit Gateway
- Site-to-SiteVPN AWS
- VPC
- AWS Resource Access Manager (AWS RAM)

Architettura Target

Tools (Strumenti)

Servizi AWS

- AWS Resource Access Manager (AWS RAM) ti aiuta a condividere in modo sicuro le tue risorse tra gli account AWS, le unità organizzative o l'intera organizzazione di AWS Organizations.
- AWS Transit Gateway è un hub centrale che collega cloud privati virtuali (VPCs) e reti locali.

Epiche

Crea un gateway di transito nell'account dei servizi di rete

Operazione	Description	Competenze richieste
Crea un gateway di transito.	Nell'account AWS in cui desideri ospitare i servizi di rete, crea un gateway di transito nella regione AWS di destinazione. Per istruzioni, consulta Creare un gateway di transito. Tenere presente quanto segue: • Seleziona Associazione predefinita alla tabella di percorso. • Seleziona Propagazione della tabella di routing predefinita.	Amministratore di rete

Connect il gateway di transito alla rete locale

Operazione	Description	Competenze richieste
Configura un dispositivo gateway per il cliente per la connessione VPN.	Il dispositivo gateway del cliente è collegato sul lato locale della connessione Site-to-Site VPN tra il gateway di transito e la rete locale. Per ulteriori informazioni, consulta Il tuo dispositivo gateway per i clienti nella documentazione di AWS Site-to-Site VPN. Identifica o avvia un dispositivo cliente locale supportato e annota il suo indirizzo IP pubblico. La configurazione della VPN viene completata più avanti in questa epopea.	Amministratore di rete
Nell'account dei servizi di rete, crea un allegato VPN al gateway di transito.	Per configurare una connessione, crea un allegato VPN per il gateway di transito. Per istruzioni, consulta gli allegati della VPN del gateway Transit.	Amministratore di rete
Configura la VPN sul dispositivo gateway del cliente nella tua rete locale.	Scarica il file di configurazione per la connessione Site-to-Site VPN associata al gateway di transito e configura le impostazioni VPN sul dispositivo gateway del cliente. Per istruzioni, consulta Scaricare il file di configurazione.	Amministratore di rete

Condividi il gateway di transito nell'account dei servizi di rete con altri account AWS o con la tua organizzazione

Operazione	Description	Competenze richieste
Nell'account di gestione AWS Organizations, attiva la condivisione.	Per condividere il gateway di transito con la tua organizzazione o con determinate unità organizzative, attiva la condivisione in AWS Organizations. Altrimenti, dovrai condividere il gateway di transito per ogni account singolarmente. Per istruzioni, consulta Abilitare la condivisione delle risorse all'interno di AWS Organizations.	Amministratore di sistema AWS
Crea la condivisione di risorse del gateway di transito nell'account dei servizi di rete.	Per consentire VPCs ad altri account AWS all'interno dell'organizzazione di connettersi al gateway di transito, nell'account dei servizi di rete, utilizza la console RAM AWS per condividere la risorsa del gateway di transito. Per istruzioni, consulta Creare una condivisione di risorse.	Amministratore di sistema AWS

Connect VPCs al gateway di transito

Operazione	Description	Competenze richieste
Crea allegati VPC in singoli account.	Negli account con cui è stato condiviso il gateway di	Amministratore di rete

Operazione	Description	Competenze richieste
	transito, crea allegati VPC del gateway di transito. Per istruzioni, consulta Creare un collegamento gateway di transito a un VPC.	
Accetta le richieste di allegati VPC.	Nell'account dei servizi di rete, accetta le richieste di allegati VPC del gateway di transito. Per istruzioni, consulta Accettare un allegato condiviso.	Amministratore di rete

Configurazione del routing

Operazione	Description	Competenze richieste
Configura i percorsi in un account individuale VPCs.	In ogni singolo account VPC, aggiungi percorsi alla rete locale e ad altre reti VPC, utilizzando il gateway di transito come destinazi one. Per istruzioni, consulta Aggiungere e rimuovere percorsi da una tabella di rotte.	Amministratore di rete
Configura i percorsi nella tabella delle rotte del gateway di transito.	Le rotte da VPCs e la connessione VPN devono essere propagate e devono apparire nella tabella delle rotte di default del gateway di transito. Se necessario, crea eventuali route statiche (un esempio sono le route	Amministratore di rete

Operazione	Description	Competenze richieste
	statiche per la connessione VPN statica) nella tabella delle rotte di default del gateway di transito. Per istruzioni, consulta Creare una route statica.	
Aggiungi le regole del gruppo di sicurezza e dell'elenco di controllo degli accessi alla rete (ACL).	Per le EC2 istanze e le altre risorse nel VPC, assicurati che le regole del gruppo di sicurezza e le regole ACL di rete consentano il traffico VPCs tra e la rete locale. Per istruzioni, consulta Controllare il traffico verso le risorse utilizzando i gruppi di sicurezza e Aggiungere ed eliminare regole da un ACL.	Amministratore di rete

Verifica la connettività

Operazione	Description	Competenze richieste
Verifica la connettività tra VPCs.	Assicurati che l'ACL di rete e i gruppi di sicurezza consentano il traffico ICMP (Internet Control Message Protocol), quindi esegui il ping dalle istanze in un VPC a un altro VPC anch'esso connesso al gateway di transito.	Amministratore di rete
Verifica la connettività tra VPCs e la rete locale.	Assicurati che le regole ACL di rete, le regole dei gruppi di sicurezza e tutti i firewall	Amministratore di rete

Operazione	Description	Competenze richieste
	consentano il traffico ICMP, quindi esegui il ping tra la rete locale e le istanze della. EC2 VPCs La comunicazione di rete deve essere avviata innanzitutto dalla rete locale per portare la connessione VPN allo stato. UP	

Risorse correlate

- Creazione di un'infrastruttura di rete AWS multi-VPC scalabile e sicura (white paper AWS)
- Utilizzo di risorse condivise (documentazione RAM AWS)
- Utilizzo dei gateway di transito (documentazione AWS Transit Gateway)

Configurare la crittografia HTTPS per Oracle JD Edwards EnterpriseOne su Oracle WebLogic utilizzando un Application Load Balancer

Thanigaivel Thirumalai, Amazon Web Services

Riepilogo

Questo modello spiega come configurare la crittografia HTTPS per l'offload SSL in Oracle JD Edwards sui carichi di lavoro Oracle. EnterpriseOne WebLogic Questo approccio crittografa il traffico tra il browser dell'utente e un sistema di bilanciamento del carico per rimuovere il carico di crittografia dai server. EnterpriseOne

Molti utenti scalano orizzontalmente il livello della macchina virtuale EnterpriseOne JAVA (JVM) utilizzando un Application Load Balancer di AWS.Il load balancer funge da unico punto di contatto per i client e distribuisce il traffico in entrata su più canali. JVMs Facoltativamente, il load balancer può distribuire il traffico su più zone di disponibilità e aumentare la disponibilità di. EnterpriseOne

Il processo descritto in questo schema configura la crittografia tra il browser e il sistema di bilanciamento del carico anziché crittografare il traffico tra il sistema di bilanciamento del carico e il. EnterpriseOne JVMs Questo approccio è denominato offloading SSL. L'offload del processo di decrittografia SSL dal server EnterpriseOne Web o dell'applicazione all'Application Load Balancer riduce il carico sul lato dell'applicazione. Dopo la terminazione SSL presso il sistema di bilanciamento del carico, il traffico non crittografato viene indirizzato all'applicazione su AWS.

Oracle JD Edwards EnterpriseOne è una soluzione ERP (Enterprise Resource Planning) per organizzazioni che producono, costruiscono, distribuiscono, forniscono assistenza o gestiscono prodotti o risorse fisiche. JD Edwards EnterpriseOne supporta vari hardware, sistemi operativi e piattaforme di database.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un ruolo AWS Identity and Access Management (IAM) che dispone delle autorizzazioni per effettuare chiamate di servizio AWS e gestire le risorse AWS
- Un certificato SSL

Versioni del prodotto

- Questo modello è stato testato con Oracle WebLogic 12c, ma è possibile utilizzare anche altre versioni.

Architecture

Esistono diversi approcci per eseguire l'offload SSL. Questo modello utilizza un Application Load Balancer e Oracle HTTP Server (OHS), come illustrato nel diagramma seguente.

Il diagramma seguente mostra il layout JVM di JD Edwards EnterpriseOne, Application Load Balancer e Java Application Server (JAS).

Tools (Strumenti)

Servizi AWS

- Gli Application Load Balancer distribuiscono il traffico delle applicazioni in entrata su più destinazioni, come Amazon Elastic Compute Cloud (EC2 istanze Amazon), in più zone di disponibilità.
- AWS Certificate Manager (ACM) ti aiuta a creare, archiviare e rinnovare certificati e chiavi SSL/TLS X.509 pubblici e privati che proteggono i tuoi siti Web e le tue applicazioni AWS.
- Amazon Route 53 è un servizio Web DNS altamente scalabile e disponibile.

Best practice

- Per le best practice ACM, consulta la documentazione ACM.

Epiche

Configurazione WebLogic e OHS

Operazione	Description	Competenze richieste
Installa e configura i componenti Oracle.	1. Installa Fusion Middlewar e Infrastructure seguendo la procedura di installaz	JDE CNC, amministratore WebLogic

Operazione	Description	Competenze richieste
	ione standard. Questo programma consente di installare e configurare un WebLogic dominio. Per istruzioni, consulta la documentazione di Oracle. 2. Installa OHS seguendo la procedura di installazione standard. Per istruzioni, consulta la documentazione di Oracle. 3. Una volta completat a l'installazione, avvia la procedura guidata di configurazione (`config.sh` file) per configurare OHS. • È possibile aggiornar e un dominio esistente o creare un nuovo dominio. Questo modello presuppone che tu stia aggiornando un dominio esistente. • Per i modelli disponibili, scegli Oracle Enterpris e Manager-Restricted JRF e Oracle HTTP Server (Restricted JRF). La selezione di queste opzioni Java Required Files (JRF) elimina la connessione a un database esterno.	

Operazione	Description	Competenze richieste
	• Per Server gestiti, cluster, modelli di server, cluster Coherence, macchine, Assegna server alle macchine, destinazioni virtuali e partizioni, accetta i valori di configurazione predefiniti e scegli Avanti per passare alla categoria successiva. • Completate i dettagli di configurazione (ad esempio, host e porta dell'amministratore, indirizzo e porta di ascolto, nome del server) per l'istanza OHS (ad esempio,). `ohs1`	

Operazione	Description	Competenze richieste
Abilita il WebLogic plugin a livello di dominio.	Il WebLogic plugin è necessario per il bilanciamento del carico. Per abilitare il plugin: 1. Accedi alla console di WebLogic amministrazione utilizzando il link: `http://<WeblogicServer>:<Adminport>/console` 2. Scegli Blocca e modifica, quindi scegli Configurazione, Applicazioni Web. 3. Scegli il WebLogic Plugin abilitato (casella di controllo o opzione a discesa). 4. Scegli Salva e attiva le modifiche.	JDE CNC, amministratore WebLogic

Operazione	Description	Competenze richieste

Modifica il file di configura zione.	Il `mod_wl_ohs.conf` file configura le richieste proxy da OHS a. WebLogic
	
	1. Modifica questo file. Si trova in:
	
	`$ORACLE_HOME/user_ projects/domains/`
	
	Esempio:
	
	`/home/oracle/Oracl e/Middleware/Oracl e_Home/user_projec ts/domains/base_do main/config/fmwcon fig/components/OHS /instances/ohs1`
	2. Aggiungi i valori WebLogic host (`WebLogicHost`) e port (`WebLogicPort`) (questo modello presuppon e localhost e port 8000).
	3. Aggiungi `WLProxySSL` e `WLProxySSLPassThro ugh` valori come segue:
	
	`<VirtualHost *:8000> <Location /jde> WLSRequest On SetHandler weblogic- handler WebLogicHost localhost WebLogicPort 8000`
		JDE CNC, amministratore WebLogic

Operazione	Description	Competenze richieste

	`WLProxySSL On`
	`WLProxySSLPassThrough On`
	`</Location>`
	`</VirtualHost>`	

Operazione	Description	Competenze richieste
Avviare OHS utilizzando Enterprise Manager.	1. Accedere a Enterprise Manager Fusion Middlewar e utilizzando il link: `http://<WeblogicSe rver>:<Adminport>/ em/` 2. In Target Navigation, in HTTP Server, seleziona l'istanza OHS (ad esempio,) . `ohs1` 3. Scegli Chiudi sessione e avvia per riavviare l'istanza OHS. 4. Una volta completata la configurazione OHS, è possibile connettersi al client EnterpriseOne HTML utilizzando il nome host del server HTTP con porta 8000 anziché il nome host del EnterpriseOne server. • Vecchio link: `http://<W ebserver>:80/jde/ owhtml` • Nuovo link: `http:// <HTTP server or web server>:8000/jde/o whtml` Se utilizzi una porta diversa dalla porta HTTP Oracle predefinita, modifica il `httpd.conf` file per	JDE CNC, amministratore WebLogic

Operazione	Description	Competenze richieste

	aggiungere un listener per quella porta in due punti:
	`#[Listen] OHS_LISTEN_PORT`
	`Listen 8000`
	e:
	`# ServerName <Weblogic Server1>:8000`	

Configurazione dell'Application Load Balancer

Operazione	Description	Competenze richieste
Configura un gruppo target.	1. Crea un gruppo target per la porta 8000 del server HTTP. 2. Registra le destinazioni nel gruppo di destinazione con la stessa porta. 3. Controlla lo stato dei bersagli per confermare che siano integri. 4. Se necessario, configura le impostazioni del controllo dello stato di salute. Per istruzioni dettagliate, consulta la documentazione di Elastic Load Balancing.	Amministratore AWS

Operazione	Description	Competenze richieste
Configura il sistema di bilanciamento del carico.	1. Crea un Application Load Balancer con attributi predefiniti e il cloud privato virtuale (VPC), i gruppi di sicurezza e le sottoreti richiesti. Per istruzioni, consulta la documentazione di Elastic Load Balancing. 2. Aggiungi una voce di listener per HTTPS 443 e inoltrala al gruppo target creato nel passaggio precedente. (Per istruzioni, consulta la documentazione di Elastic Load Balancing). Un listener HTTPS richiede un certificato SSL. Puoi scegliere un certificato da ACM o caricarne uno. 3. Per entrambi gli ascoltato ri, abilita la persistenza seguendo le istruzioni nella documentazione di Elastic Load Balancing.	Amministratore AWS
Aggiungi un record Route 53 (DNS).	(Facoltativo) Puoi aggiunger e un record DNS Amazon Route 53 per il sottodominio. Questo record indicherebbe il tuo Application Load Balancer. Per istruzioni, consulta la documentazione di Route 53.	Amministratore AWS

risoluzione dei problemi

Problema	Soluzione
Il server HTTP non viene visualizzato.	Se il server HTTP non viene visualizzato nell'elenco di Target Navigation sulla console di Enterprise Manager, attenersi alla seguente procedura: 1. In WebLogic Dominio, Amministrazione, scegli Istanze OHS. 2. Scegli Crea per creare una nuova istanza OHS. 3. Fornisci un nome di istanza, quindi scegli OK per creare l'istanza. Una volta creata l'istanza e attivate le modifiche , potrai vedere il server HTTP nel pannello Target Navigation.

Risorse correlate

Documentazione AWS

- Application Load Balancer
- Utilizzo delle zone ospitate pubbliche
- Utilizzo delle zone ospitate private

Documentazione Oracle:

- Panoramica del plug-in proxy di Oracle WebLogic Server
- Installazione WebLogic del server utilizzando l'Infrastructure Installer
- Installazione e configurazione di Oracle HTTP Server

Connect ai dati e ai piani di controllo dell'Application Migration Service tramite una rete privata

Dipin Jain e Mike Kuznetsov, Amazon Web Services

Riepilogo

Questo modello spiega come è possibile connettersi a un piano AWS Application Migration Service dati e a un piano di controllo su una rete privata e protetta utilizzando gli endpoint VPC dell'interfaccia.

Application Migration Service è una soluzione altamente automatizzata lift-and-shift (rehost) che semplifica, accelera e riduce i costi di migrazione delle applicazioni verso. AWS Consente alle aziende di reospitare un gran numero di server fisici, virtuali o cloud senza problemi di compatibilità, interruzioni delle prestazioni o lunghi intervalli di tempo. Il servizio di migrazione delle applicazioni è disponibile presso. Console di gestione AWS Ciò consente una perfetta integrazione con altri Servizi AWS AWS CloudTrail, come Amazon CloudWatch e AWS Identity and Access Management (IAM).

È possibile connettersi da un data center di origine a un piano dati, ovvero a una sottorete che funge da area di staging per la replica dei dati nel VPC di destinazione, tramite una connessione privata utilizzando i servizi o il peering VPC in Application Migration Site-to-Site VPN Service. AWS Direct ConnectÈ inoltre possibile utilizzare l'interfaccia VPC endpoint powered by AWS PrivateLink per connettersi a un piano di controllo dell'Application Migration Service su una rete privata.

Prerequisiti e limitazioni

Prerequisiti

- Subnet dell'area di staging: prima di configurare Application Migration Service, create una sottorete da utilizzare come area di staging per i dati replicati dai server di origine AWS (ovvero, un piano dati). È necessario specificare questa sottorete nel modello Replication Settings quando si accede per la prima volta alla console di Application Migration Service. È possibile sovrascrivere questa sottorete per server di origine specifici nel modello Replication Settings. Sebbene sia possibile utilizzare una sottorete esistente nel sistema Account AWS, si consiglia di creare una nuova sottorete dedicata a questo scopo.
- Requisiti di rete: i server di replica avviati da Application Migration Service nella sottorete dell'area di staging devono essere in grado di inviare dati all'endpoint dell'API Application Migration Service all'indirizzo`https://mgn.<region>.amazonaws.com/`, dov'è il codice su cui si `<region>` sta

eseguendo la replica (ad esempio,). Regione AWS `https://mgn.us-east-1.amazonaws.com` Il servizio Amazon Simple Storage Service (Amazon S3) Simple Storage URLs Service (Amazon S3) è necessario per scaricare il software Application Migration Service.

- Il programma di installazione di AWS Replication Agent dovrebbe avere accesso all'URL del bucket Amazon Simple Storage Service (Amazon S3) del Regione AWS bucket utilizzato con Application Migration Service.
- La sottorete dell'area di staging dovrebbe avere accesso ad Amazon S3.
- I server di origine su cui è installato il AWS Replication Agent devono essere in grado di inviare dati ai server di replica nella sottorete dell'area di staging e all'endpoint dell'API Application Migration Service all'indirizzo. `https://mgn.<region>.amazonaws.com/`

La tabella seguente elenca le porte richieste.

Origine	Destinazione	Porta	Per ulteriori informazioni, consultare la pagina
Centro dati di origine	Servizio Amazon S3 URLs	443 (TCP)	Comunicazione tramite porta TCP 443
Centro dati di origine	Regione AWS-indirizzo di console specifico per Application Migration Service	443 (TCP)	Comunicazione tra i server di origine e Application Migration Service tramite la porta TCP 443
Centro dati di origine	Sottorete dell'area di staging	1500 (TCP)	Comunicazione tra i server di origine e la sottorete dell'area di staging tramite la porta TCP 1500
Sottorete dell'area di staging	Regione AWS-indirizzo di console specifico per Application Migration Service	443 (TCP)	Comunicazione tra la sottorete dell'area di staging e Application Migration Service

			tramite la porta TCP 443
Area di staging (sottorete)	Servizio Amazon S3 URLs	443 (TCP)	Comunicazione tramite porta TCP 443
Area di staging (sottorete)	Endpoint Amazon Elastic Compute Cloud (Amazon EC2) della sottorete Regione AWS	443 (TCP)	Comunicazione tramite la porta TCP 443

Limitazioni di

Il servizio di migrazione delle applicazioni non è attualmente disponibile in tutti Regioni AWS i sistemi operativi.

- Supportato Regioni AWS
- Sistemi operativi supportati

Architecture

Il diagramma seguente illustra l'architettura di rete per una migrazione tipica. Per ulteriori informazioni su questa architettura, vedere la documentazione di Application Migration Service e il video sull'architettura del servizio Application Migration Service e sull'architettura di rete.

La seguente visualizzazione dettagliata mostra la configurazione degli endpoint VPC dell'interfaccia nell'area di staging VPC per connettere Amazon S3 e Application Migration Service.

Tools (Strumenti)

- AWS Application Migration Servicesemplifica, accelera e riduce i costi di rehosting delle applicazioni su. AWS
- Gli endpoint VPC di interfaccia consentono di connettersi a servizi alimentati da AWS PrivateLink senza richiedere un gateway Internet, un dispositivo NAT, una connessione VPN o una connessione. AWS Direct Connect Le istanze nel VPC non richiedono indirizzi IP pubblici per

comunicare con le risorse nel servizio. Il traffico tra il VPC e gli altri servizi rimane all'interno della rete Amazon.

Epiche

Crea endpoint per Application Migration Service EC2, Amazon e Amazon S3

Operazione	Description	Competenze richieste
Configurare l'endpoint di interfaccia per Application Migration Service.	Il data center di origine e l'area di staging (VPC) si connettono privatamente al piano di controllo dell'Application Migration Service tramite l'endpoint di interfaccia creato nell'area di staging di destinazione (VPC). Per creare l'endpoint: 1. Apri la console Amazon Virtual Private Cloud (Amazon VPC). 2. Nel riquadro di navigazione, seleziona Endpoints (Endpoint), Create Endpoint (Crea endpoint). 3. Per Service category (Categoria servizio), scegli Servizi AWS. 4. Per il nome del servizio, inserisci `com.amazonaws.<region>.mgn`. Per Tipo, scegliete Interfaccia. 5. Per VPC, seleziona un VPC dell'area di staging	Responsabile della migrazione

Operazione	Description	Competenze richieste
	di destinazione per creare l'endpoint. 6. Per Subnet, seleziona le sottoreti in cui creare le interfacce di rete degli endpoint. 7. Per attivare il DNS privato per l'endpoint di interfaccia, nella sezione Impostazioni aggiuntive, seleziona Abilita nome DNS. 8. Seleziona un gruppo di sicurezza che consenta l'accesso dalla sottorete VPC dell'area di staging tramite TCP 443. 9. Seleziona Crea endpoint. Per ulteriori informazioni, consulta Accedere e Servizio AWS utilizzare un endpoint VPC di interfaccia nella documentazione di Amazon VPC.	

Operazione	Description	Competenze richieste
Configura l'endpoint di interfaccia per Amazon EC2.	L'area di staging VPC si connette privatamente all'API EC2 Amazon tramite l'endpoint di interfaccia creato nell'area di staging di destinazione VPC. Per creare l'endpoint, segui le istruzioni fornite nella storia precedente. • Per il nome del servizio, immetterecom.amazonaws.<region>.ec2 . Per Tipo, scegliete Interfaccia. • Il gruppo di sicurezza deve consentire il traffico HTTPS in entrata dalla sottorete VPC dell'area di staging sulla porta 443. • Nella sezione Impostazioni aggiuntive, seleziona Abilita nome DNS.	Responsabile della migrazione

Operazione	Description	Competenze richieste
Configura l'endpoint di interfaccia per Amazon S3.	Il data center di origine e l'area di staging (VPC) si connettono privatamente all'API Amazon S3 tramite l'endpoint di interfaccia creato nel VPC dell'area di staging di destinazione. Per creare l'endpoint, segui le istruzioni fornite nella prima storia. • Per Nome del servizio, immetterecom.amazonaws.<region>.s3 . Per Tipo, scegliete Interfaccia. • Il gruppo di sicurezza VPC deve consentire il traffico HTTPS in entrata dalla sottorete VPC dell'area di staging sulla porta 443. • Nella sezione Impostazioni aggiuntive, deselezionate Abilita nome DNS. Gli endpoint dell'interfaccia Amazon S3 non supportano nomi DNS privati. **Note** Si utilizza un endpoint di interfaccia perché le connessioni degli endpoint gateway non possono essere	Responsabile della migrazione

Operazione	Description	Competenze richieste
	estese al di fuori di un VPC. (Per i dettagli, consulta la AWS PrivateLink documentazione.)	
Configura l'endpoint Amazon S3 Gateway.	Durante la fase di configurazione, il server di replica deve connettersi a un bucket S3 per scaricare gli aggiornamenti software del AWS Replication Server. Tuttavia, gli endpoint dell'interfaccia Amazon S3 non supportano nomi DNS privati e non è possibile fornire un nome DNS di endpoint Amazon S3 a un server di replica. Per mitigare questo problema, crei un endpoint gateway Amazon S3 nel VPC a cui appartiene la sottorete dell'area di staging e aggiorna le tabelle di routing della sottorete di staging con le route pertinenti. Per ulteriori informazioni, consulta Creare un endpoint gateway nella documentazione. AWS PrivateLink	Amministratore cloud

Operazione	Description	Competenze richieste
Configura il DNS locale per risolvere i nomi DNS privati per gli endpoint.	Gli endpoint di interfaccia per Application Migration Service e Amazon EC2 dispongon o di nomi DNS privati che possono essere risolti nel VPC. Tuttavia, devi anche configurare i server locali per risolvere i nomi DNS privati per questi endpoint di interfacc ia. Esistono diversi modi per configurare questi server. In questo modello, abbiamo testato questa funzionalità inoltrando le query DNS locali all'endpoint in Amazon Route 53 Resolver entrata nell'area di staging VPC. Per ulteriori informazioni, consulta Risoluzione delle query DNS tra e la rete nella documenta zione di Route 53. VPCs	Ingegnere della migrazione

Connettersi al piano di controllo dell'Application Migration Service tramite un collegamento privato

Operazione	Description	Competenze richieste
Installare AWS Replicati on Agent utilizzando AWS PrivateLink.	1. Scarica il AWS Replicati on Agent in un bucket S3 privato nella regione di destinazione. 2. Accedere ai server di origine da migrare. Il	Tecnico di migrazione

Operazione	Description	Competenze richieste
	programma di installazione AWS di Replication Agent richiede l'accesso di rete all'Application Migration Service e agli endpoint Amazon S3. Poiché la tua rete locale non è aperta agli endpoint pubblici di Application Migration Service e Amazon S3, devi installare l'agente con l'aiuto degli endpoint di interfaccia che hai creato nei passaggi precedenti utilizzando. AWS PrivateLink Ecco un esempio per Linux: 1. Scarica l'agente utilizzando il comando: `wget -O ./aws-replication-installer-init.py \ https://aws-application-migration-service-<aws_region>.bucket.<s3-endpoint-DNS-name>/latest/linux/aws-replication-installer-init.py` Ad esempio, se il nome DNS dell'endpoint dell'interfaccia Amazon S3	

Operazione	Description	Competenze richieste
	`vpce-009c8b07adb052a11-qgf8q50y.s3.us-west-1.vpce.amazonaws.com` è and Regione AWS the `us-west-1` is, dovresti usare il comando: `wget -O ./aws-replication-installer-init.py \ https://aws-application-migration-service-us-west-1.bucket.vpce-009c8b07adb052a11-qgf8q50y.s3.us-west-1.vpce.amazonaws.com/latest/linux/aws-replication-installer-init.py` **Note** `bucket`è una parola chiave statica che devi aggiungere prima del nome DNS dell'endpoint dell'interfaccia Amazon S3. Per ulteriori informazioni, consulta la Documentazione di Amazon S3.	

Operazione	Description	Competenze richieste

	2. Installa l'agente:
	• Se hai selezionato Abilita nome DNS quando hai creato un endpoint di interfaccia per Application Migration Service, esegui il comando:
	`sudo python3 aws-replication-installer-init.py \ --region <aws_region> \ --aws-access-key-id <access-key> \ --aws-secret-access-key <secret-key> \ --no-prompt \ --s3-endpoint <s3-endpoint-DNS-name>`
	• Se non hai selezionato Abilita nome DNS quando hai creato l'endpoint di interfaccia per Application Migration Service, esegui il comando:
	`sudo python3 aws-replication-installer-init.py \ --region <aws_region> \ --aws-access-key-id <access-key> \`	

Operazione	Description	Competenze richieste
	`--aws-secret-access-key <secret-key> \ --no-prompt \ --s3-endpoint <s3-endpoint-DNS-name> \ --endpoint <mgn-endpoint-DNS-name>` Per ulteriori informazioni, vedere le istruzioni di installazione di AWS Replication Agent nella documentazione di Application Migration Service. Dopo aver stabilito la connessione con Application Migration Service e installato AWS Replication Agent, segui le istruzioni nella documentazione di Application Migration Service per migrare i server di origine al VPC e alla sottorete di destinazione.	

Risorse correlate

Documentazione del servizio di migrazione delle applicazioni

- Concetti
- Workflow di migrazione
- Guida rapida di avvio

- DOMANDE FREQUENTI
- Risoluzione dei problemi

Altre risorse

- Rehosting delle applicazioni in un'architettura multi-account utilizzando gli endpoint dell'interfaccia VPC (guida Prescriptive Guidance) AWS AWS
- AWS Application Migration Service — Un'introduzione tecnica (procedura dettagliata di formazione e certificazione)AWS
- AWS Application Migration Service architettura e architettura di rete (video)

Informazioni aggiuntive

Risoluzione dei problemi relativi alle installazioni AWS di Replication Agent sui server Linux

Se ricevi un errore gcc su un server Amazon Linux, configura l'archivio dei pacchetti e usa il seguente comando:

```
## sudo yum groupinstall "Development Tools"
```

Crea oggetti Infoblox utilizzando risorse CloudFormation personalizzate AWS e Amazon SNS

Tim Sutton, Amazon Web Services

Riepilogo

Avviso: non AWS Cloud9 è più disponibile per i nuovi clienti. I clienti esistenti di AWS Cloud9 possono continuare a utilizzare il servizio normalmente. Ulteriori informazioni

Infoblox Domain Name System (DNS), Dynamic Host Configuration Protocol (DHCP) e gestione degli indirizzi IP (Infoblox DDI) consentono di centralizzare e controllare in modo efficiente un ambiente ibrido complesso. Con Infoblox DDI, è possibile scoprire e registrare tutte le risorse di rete in un database autorevole di gestione degli indirizzi IP (IPAM), oltre a gestire il DNS in locale e sul cloud Amazon Web Services (AWS) utilizzando le stesse appliance.

Questo modello descrive come utilizzare una risorsa CloudFormation personalizzata AWS per creare oggetti Infoblox (ad esempio record DNS o oggetti IPAM) chiamando l'API WAPI di Infoblox. Per ulteriori informazioni sulla WAPI di Infoblox, consulta la documentazione WAPI nella documentazione di Infoblox.

Utilizzando l'approccio di questo modello, puoi ottenere una visione unificata dei record DNS e delle configurazioni IPAM per i tuoi ambienti AWS e locali, oltre a rimuovere i processi manuali che creano record e forniscono le tue reti. È possibile utilizzare l'approccio di questo pattern per i seguenti casi d'uso:

- Aggiungere un record A dopo aver creato un'istanza Amazon Elastic Compute Cloud (Amazon EC2)
- Aggiungere un record CNAME dopo aver creato un Application Load Balancer
- Aggiungere un oggetto di rete dopo aver creato un cloud privato virtuale (VPC)
- Fornire l'intervallo di rete successivo e utilizzare tale intervallo per creare sottoreti

È inoltre possibile estendere questo modello e utilizzare altre funzionalità del dispositivo Infoblox, come l'aggiunta di diversi tipi di record DNS o la configurazione di Infoblox vDiscovery.

Il modello utilizza un hub-and-spoke design in cui l'hub richiede la connettività all'appliance Infoblox sul cloud AWS o in locale e utilizza AWS Lambda per chiamare l'API Infoblox. Lo spoke si trova nello

stesso account o in un altro account della stessa organizzazione in AWS Organizations e richiama la funzione Lambda utilizzando una risorsa CloudFormation personalizzata AWS.

Prerequisiti e limitazioni

Prerequisiti

- Un'appliance o una griglia Infoblox esistente, installata sul cloud AWS, in locale o entrambi, e configurata con un utente amministratore in grado di amministrare azioni IPAM e DNS. Per ulteriori informazioni a riguardo, consulta Informazioni sugli account di amministrazione nella documentazione di Infoblox.
- Una zona autorevole DNS esistente a cui si desidera aggiungere record sull'appliance Infoblox. Per ulteriori informazioni su questo argomento, vedere Configurazione delle zone autoritative nella documentazione di Infoblox.
- Due account AWS attivi in AWS Organizations. Un account è l'account hub e l'altro account è l'account spoke.
- Gli account hub and spoke devono trovarsi nella stessa regione AWS.
- Il VPC dell'account hub deve connettersi all'appliance Infoblox, ad esempio utilizzando AWS Transit Gateway o il peering VPC.
- AWS Serverless Application Model (AWS SAM), installato e configurato localmente con AWS Cloud9 o AWS. CloudShell
- I `ClientTest.yaml` file `Infoblox-Hub.zip` and (allegati), scaricati nell'ambiente locale che contiene AWS SAM.

Limitazioni

- Il token di servizio della risorsa CloudFormation personalizzata AWS deve provenire dalla stessa regione in cui viene creato lo stack. Ti consigliamo di utilizzare un account hub in ogni regione, anziché creare un argomento Amazon Simple Notification Service (Amazon SNS) in una regione e richiamare la funzione Lambda in un'altra regione.

Versioni del prodotto

- Infoblox WAPI versione 2.7

Architecture

I seguenti diagrammi mostrano il flusso di lavoro di questo modello.

Il diagramma mostra i seguenti componenti per la soluzione di questo pattern:

1. Le risorse CloudFormation personalizzate AWS ti consentono di scrivere logiche di provisioning personalizzate nei modelli che AWS CloudFormation esegue quando crei, aggiorni o elimini stack. Quando crei uno stack, AWS CloudFormation invia una `create` richiesta a un argomento SNS monitorato da un'applicazione in esecuzione su un' EC2 istanza.
2. La notifica Amazon SNS proveniente dalla risorsa CloudFormation personalizzata AWS è crittografata tramite una chiave AWS Key Management Service (AWS KMS) specifica e l'accesso è limitato agli account dell'organizzazione in Organizations. L'argomento SNS avvia la risorsa Lambda che chiama l'API WAPI di Infoblox.
3. Amazon SNS richiama le seguenti funzioni Lambda che utilizzano l'URL WAPI Infoblox, il nome utente e la password AWS Secrets Manager Amazon Resource Names () come variabili di ambiente: ARNs
   - `dnsapi.lambda_handler`— Riceve `DNSName` i `DNSType` `DNSValue` valori e dalla risorsa CloudFormation personalizzata AWS e li utilizza per creare record DNS A e CNAMES.
   - `ipaddr.lambda_handler`— Riceve i `Network Name` valori `VPCCIDRType`,`SubnetPrefix`, e dalla risorsa CloudFormation personalizzata AWS e li utilizza per aggiungere i dati di rete al database IPAM di Infoblox o fornire alla risorsa personalizzata la prossima rete disponibile che può essere utilizzata per creare nuove sottoreti.
   - `describeprefixes.lambda_handler`— Richiama l'API `describe_managed_prefix_lists` AWS utilizzando il `"com.amazonaws."+Region +".s3"` filtro per recuperare i dati richiesti. `prefix ID`

   > **Important**
   >
   > Queste funzioni Lambda sono scritte in Python e sono simili tra loro ma chiamano in modo diverso. APIs
4. È possibile implementare la griglia Infoblox come appliance di rete fisiche, virtuali o basate sul cloud. Può essere distribuito in locale o come appliance virtuale utilizzando una serie di hypervisor, tra cui VMware ESXi Microsoft Hyper-V, Linux KVM e Xen. Puoi anche distribuire la griglia Infoblox sul cloud AWS con un'Amazon Machine Image (AMI).

5. Il diagramma mostra una soluzione ibrida per la griglia Infoblox che fornisce DNS e IPAM alle risorse sul cloud AWS e in locale.

Stack tecnologico

- AWS CloudFormation
- IAM
- AWS KMS
- AWS Lambda
- AWS SAM
- AWS Secrets Manager
- Amazon SNS
- Amazon VPC

Tools (Strumenti)

- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account e regioni AWS.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- AWS Organizations è un servizio di gestione degli account che ti aiuta a consolidare più account AWS in un'organizzazione da creare e gestire centralmente.
- AWS Secrets Manager ti aiuta a sostituire le credenziali codificate nel codice, comprese le password, con una chiamata API a Secrets Manager per recuperare il segreto a livello di codice.
- AWS Serverless Application Model (AWS SAM) Serverless Application Model (AWS SAM) è un framework open source che ti aiuta a creare applicazioni serverless nel cloud AWS.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.

- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito. Questa rete virtuale è simile a una rete tradizionale che gestiresti nel tuo data center, con i vantaggi dell'utilizzo dell'infrastruttura scalabile di AWS.

Codice

Puoi utilizzare il CloudFormation modello AWS di `ClientTest.yaml` esempio (allegato) per testare l'hub Infoblox. Puoi personalizzare il CloudFormation modello AWS per includere le risorse personalizzate dalla tabella seguente.

Crea un record A utilizzando la risorsa personalizzata Infoblox Spoke	Valori restituiti:
	`infobloxref` — Riferimenti Infoblox
	Risorsa di esempio:
	`ARECORDCustomResource:`
	`  Type: "Custom::InfobloxAPI"`
	`  Properties:`
	`  ServiceToken: !Sub arn:aws:sns:${AWS::Region}:${HubAccountID}:RunInfobloxDNSFunction`
	`  DNSName: 'arecordtest.company.com'`
	`  DNSType: 'ARecord'`
	`  DNSValue: '10.0.0.1'`

Crea un record CNAME utilizzando la risorsa personalizzata Infoblox spoke	Valori restituiti: `infobloxref` — Riferimenti Infoblox Risorsa di esempio: `CNAMECustomResource:`

```
  Type: "Custom::InfobloxAPI"

  Properties:

    ServiceToken: !Sub arn:aws:sns:
${AWS::Region}:${HubAccountID}:Ru
nInfoblox

    DNSFunction

    DNSName: 'cnametest.company.com'

    DNSType: 'cname'

    DNSValue: 'aws.amazon.com'
```

Crea un oggetto di rete utilizzando la risorsa personalizzata Infoblox spoke

Valori restituiti:

`infobloxref` — Riferimenti Infoblox

`network`— Intervallo di rete (uguale a) `VPCCIDR`

Risorsa di esempio:

```
VPCCustomResource:

  Type: 'Custom::InfobloxAPI'

  Properties:

    ServiceToken: !Sub  arn:aws:sns:
${AWS::Region}:${HubAccountID}:Ru
nInfobloxNextSubnetFunction

    VPCCIDR: !Ref VpcCIDR

    Type: VPC

    NetworkName: My-VPC
```

Recupera la prossima sottorete disponibile utilizzando la risorsa personalizzata Infoblox spoke

Valori restituiti:

`infobloxref` — Riferimenti Infoblox

`network` — L'intervallo di rete della sottorete

Risorsa di esempio:

```
Subnet1CustomResource:

  Type: 'Custom::InfobloxAPI'

  DependsOn: VPCCustomResource

  Properties:

    ServiceToken: !Sub  arn:aws:sns:
${AWS::Region}:${HubAccountID}:Ru
nInfobloxNextSubnetFunction

    VPCCIDR: !Ref VpcCIDR

    Type: Subnet

    SubnetPrefix: !Ref SubnetPrefix

NetworkName: My-Subnet
```

Epiche

Crea e configura il VPC dell'account hub

Operazione	Description	Competenze richieste
Creare un VPC con una connessione all'appliance Infoblox.	Accedi alla Console di gestione AWS per il tuo account hub e crea un VPC seguendo i passaggi della distribuzione di riferimento Amazon VPC sulla distribuz	Amministratore di rete, amministratore di sistema

Operazione	Description	Competenze richieste
	ione di riferimento AWS Cloud Quick Start da AWS Quick Starts. **Important** Il VPC deve disporre di connettività HTTPS all'appliance Infoblox e si consiglia di utilizzar e una sottorete privata per questa connessio ne.	

Operazione	Description	Competenze richieste
(Facoltativo) Crea gli endpoint VPC per le sottoreti private.	Gli endpoint VPC forniscono connettività ai servizi pubblici per le sottoreti private. Sono richiesti i seguenti endpoint: • Un endpoint gateway per Amazon Simple Storage Service (Amazon S3) per consentire a Lambda di comunicare con AWS CloudFormation • Un endpoint di interfaccia per Secrets Manager per abilitare la connettività con Secrets Manager • Un endpoint di interfaccia per AWS KMS per consentire la crittografia dell'argomento SNS e del segreto di Secrets Manager Per ulteriori informazioni sulla creazione di endpoint per sottoreti private, consulta Endpoint VPC nella documentazione di Amazon VPC.	Amministratore di rete, amministratore di sistema

Implementa l'hub Infoblox

Operazione	Description	Competenze richieste
Crea il modello AWS SAM.	1. Esegui il `unzip Infoblox-Hub.zip` comando nell'ambiente che contiene AWS SAM. 2. Esegui il `cd Hub/` comando per cambiare la tua directory nella `Hub` directory. 3. Esegui il `sam build` comando per elaborare il file modello AWS SAM, il codice dell'applicazione e qualsiasi file e dipendenz a specifici del linguaggio. Il `sam build` comando copia anche gli artefatti della build nel formato e nella posizione previsti per la seguente storia.	Sviluppatore, amministratore di sistema
Implementa il modello AWS SAM.	Il `sam deploy` comando prende i parametri richiesti e li salva nel `samconfig.toml` file, archivia il CloudForm ation modello AWS e le funzioni Lambda in un bucket S3, quindi distribuisce il CloudFormation modello AWS nell'account dell'hub. Il seguente codice di esempio mostra come distribuire il modello AWS SAM:	Sviluppatore, amministratore di sistema

Operazione	Description	Competenze richieste

	`$ sam deploy --guided`
	
	`Configuring SAM deploy`
	`==========`
	`==`
	`Looking for config file [samconfig.toml] : Found`
	`Reading default arguments : Success`
	`Setting default arguments for 'sam deploy'`
	`==========`
	`==========`
	`Stack Name [Infoblox-Hub]:`
	`AWS Region [eu-west-1]:`
	`Parameter InfobloxUsername:`
	`Parameter InfobloxPassword:`
	`Parameter InfobloxIPAddress [xxx.xxx.xx.xxx]:`
	`Parameter AWSOrganisationID [o-xxxxxxxxx]:`
	`Parameter VPCID [vpc-xxxxxxxxx]:`
	`Parameter VPCCIDR [xxx.xxx.xxx.xxx/16]:`
	`Parameter VPCSubnetID1 [subnet-xxx]:`
	`Parameter VPCSubnetID2 [subnet-xxx]:`	

Operazione	Description	Competenze richieste
	Parameter VPCSubnetID3 [subnet-xxx]: Parameter VPCSubnetID4 []: #Shows you resources changes to be deployed and require a 'Y' to initiate deploy Confirm changes before deploy [Y/n]: y #SAM needs permission to be able to create roles to connect to the resources in your template Allow SAM CLI IAM role creation [Y/n]: n Capabilities [['CAPABILITY_NAMED_IAM']]: Save arguments to configuration file [Y/n]: y SAM configuration file [samconfig.toml]: SAM configuration environment [default]: **Important** È necessario utilizzare l'`--guided`opzione ogni volta perché le credenziali di accesso di Infoblox non sono	

Operazione	Description	Competenze richieste
	memorizzate nel file. `samconfig.toml`	

Risorse correlate

- Guida introduttiva all' WAPIs utilizzo di Postman (Infoblox Blog)
- Provisioning di VNIOS per AWS utilizzando il modello BYOL (documentazione Infoblox)
- quickstart-aws-vpc(GitHub repo)
- describe_managed_prefix_lists (documentazione dell'SDK AWS per Python)

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Crea un'architettura IPAM gerarchica e multiregionale utilizzando Terraform AWS

Donny Schreiber, Amazon Web Services

Riepilogo

La gestione degli indirizzi IP (IPAM) è un componente fondamentale della gestione della rete e diventa sempre più complessa man mano che le organizzazioni scalano la propria infrastruttura cloud. Senza un IPAM adeguato, le organizzazioni rischiano conflitti di indirizzi IP, spreco di spazio di indirizzamento e risoluzione di problemi complessi che possono portare a interruzioni e tempi di inattività delle applicazioni. Questo modello dimostra come implementare una soluzione IPAM completa per ambienti aziendali utilizzando Terraform. AWS HashiCorp Aiuta le organizzazioni a creare un'architettura IPAM gerarchica e multiregionale che facilita la gestione centralizzata degli indirizzi IP in tutta l'organizzazione. Account AWSAWS

Questo modello ti aiuta a implementare Amazon VPC IP Address Manager con una sofisticata gerarchia di pool a quattro livelli: pool di primo livello, pool regionali, pool di unità aziendali e pool specifici per ambiente. Questa struttura supporta una corretta governance degli indirizzi IP, consentendo al contempo la delega della gestione degli IP ai team appropriati all'interno dell'organizzazione. La soluzione utilizza AWS Resource Access Manager (AWS RAM) per condividere senza problemi i pool di IP Address Manager in tutta l'organizzazione. AWS RAM centralizza e standardizza le specifiche IPAM, su cui i team possono basarsi per tutti gli account gestiti.

Questo modello può aiutarti a raggiungere i seguenti obiettivi:

- Automatizza l'allocazione degli indirizzi IP tra Regioni AWS unità aziendali e ambienti.
- Applica le politiche di rete organizzative attraverso la convalida programmatica.
- Scala l'infrastruttura di rete in modo efficiente man mano che i requisiti aziendali si evolvono.
- Riduci il sovraccarico operativo attraverso la gestione centralizzata degli spazi degli indirizzi IP.
- Accelera le implementazioni dei carichi di lavoro nativi sul cloud con l'allocazione self-service della gamma CIDR.
- Previeni i conflitti di indirizzo attraverso controlli e convalide basati su policy.

Prerequisiti e limitazioni

Prerequisiti

- Uno o più Account AWS, gestiti come organizzazione in. AWS Organizations
- Un hub di rete o un account di gestione della rete che fungerà da amministratore delegato di IP Address Manager.
- AWS Command Line Interface (AWS CLI), installato e configurato.
- Terraform versione 1.5.0 o successiva, installata.
- AWS Provider per Terraform, configurato.
- Autorizzazioni per gestire IP Address Manager e cloud privati virtuali (VPCs), configurati in AWS Identity and Access Management (IAM). AWS RAM

Limitazioni

- IP Address Manager è soggetto a quote di servizio. La quota di servizio predefinita per i pool è 50 per ambito. L'esecuzione di questa distribuzione per 6 regioni, 2 unità aziendali e 4 ambienti creerebbe 67 pool. Pertanto, potrebbe essere necessario aumentare la quota.
- La modifica o l'eliminazione dei pool di IP Address Manager dopo l'allocazione delle risorse può causare problemi di dipendenza. È necessario rilasciare l'allocazione prima di poter eliminare il pool.
- In IP Address Manager, il monitoraggio delle risorse può subire un leggero ritardo nel riflettere le modifiche alle risorse. Questo ritardo può essere di circa 20 minuti.
- IP Address Manager non può applicare automaticamente l'unicità dell'indirizzo IP in diversi ambiti.
- I tag personalizzati devono rispettare le migliori pratiche di etichettatura.AWS Ad esempio, ogni chiave deve essere unica e non può iniziare con. `aws:`
- Esistono considerazioni e limitazioni quando si integra IP Address Manager con account esterni all'organizzazione.

Architecture

Architettura Target

Configurazione e gerarchia dei pool di IP Address Manager

Il diagramma seguente mostra i costrutti logici dell'architettura di destinazione. Un ambito è il contenitore di livello più alto in IP Address Manager. Ogni ambito rappresenta lo spazio degli indirizzi IP per una singola rete. I pool sono raccolte di intervalli di indirizzi IP contigui (o intervalli CIDR) all'interno dell'ambito. I pool consentono di organizzare gli indirizzi IP in base alle esigenze di routing e sicurezza. Questo diagramma mostra quattro livelli gerarchici di pool: un pool di primo livello, pool regionali, pool di unità aziendali e pool di ambienti.

Questa soluzione stabilisce una chiara gerarchia di pool di IP Address Manager:

1. Il pool di primo livello comprende l'intero spazio di indirizzi IP dell'organizzazione, ad esempio. `10.176.0.0/12`
2. I pool regionali sono destinati ad allocazioni specifiche della regione, ad esempio per. `10.176.0.0/15` `us-east-1`
3. I pool di unità aziendali sono allocazioni specifiche del dominio all'interno di ciascuno di essi. Regione AWS Ad esempio, l'unità operativa finanziaria della regione potrebbe avere. `us-east-1` `10.176.0.0/16`
4. I pool di ambienti sono allocazioni specifiche per diversi ambienti. Ad esempio, l'unità operativa finanziaria della `us-east-1` regione potrebbe disporre di un ambiente `10.176.0.0/18` di produzione.

Questa topologia di distribuzione distribuisce geograficamente le risorse di IP Address Manager mantenendo il controllo centralizzato. Le sue caratteristiche sono le seguenti:

- IP Address Manager viene distribuito in un unico sistema primario Regione AWS.
- Le regioni aggiuntive vengono registrate come aree operative, in cui IP Address Manager può gestire le risorse.
- Ogni regione operativa riceve un pool di indirizzi dedicato dal pool di livello superiore.
- Le risorse in tutte le regioni operative sono gestite centralmente tramite IP Address Manager nella regione principale.
- Ogni pool regionale ha una proprietà locale legata alla relativa regione per aiutarti ad allocare correttamente le risorse.

Convalida avanzata dell'intervallo CIDR

Questa soluzione è progettata per impedire l'implementazione di configurazioni non valide. Quando si distribuiscono i pool tramite Terraform, durante la fase del piano Terraform viene convalidato quanto segue:

- Verifica che tutti gli intervalli CIDR dell'ambiente siano contenuti negli intervalli CIDR dell'unità aziendale principale
- Conferma che tutti gli intervalli CIDR delle unità aziendali sono contenuti negli intervalli CIDR regionali principali
- Verifica che tutti gli intervalli CIDR regionali siano contenuti negli intervalli CIDR di primo livello
- Verifica la presenza di intervalli CIDR sovrapposti all'interno dello stesso livello gerarchico
- Convalida la corretta mappatura degli ambienti nelle rispettive unità aziendali

Allocazione dell'intervallo CIDR

Il diagramma seguente mostra un esempio di come gli sviluppatori o gli amministratori possono creare nuovi indirizzi IP VPCs e allocarli dai livelli del pool.

Il diagramma mostra il flusso di lavoro seguente:

1. Tramite il Console di gestione AWS AWS CLI, o tramite infrastructure as code (IaC), uno sviluppatore o un amministratore richiede il successivo intervallo CIDR disponibile nel pool di `AY3` ambienti.
2. IP Address Manager alloca il successivo intervallo CIDR disponibile in quel pool al `AY3-4` VPC. Questo intervallo CIDR non può più essere utilizzato.

Automazione e scalabilità

Questa soluzione è progettata per la scalabilità come segue:

- Espansione regionale: aggiungi nuove regioni estendendo la configurazione Terraform con ulteriori voci del pool regionale.
- Crescita delle unità aziendali: supporta nuove unità aziendali aggiungendole alla mappa di configurazione della BU.
- Flessibilità dell'ambiente: configura diversi tipi di ambiente, come lo sviluppo o la produzione, in base alle esigenze organizzative.

- Supporto per più account: condividi i pool tra tutti gli account dell'organizzazione tramite AWS RAM.
- Provisioning VPC automatizzato: integrazione con i flussi di lavoro di provisioning VPC per automatizzare l'allocazione degli intervalli CIDR.

La struttura gerarchica consente inoltre diverse scale di delega e controllo, come le seguenti:

- Gli amministratori di rete potrebbero gestire i pool di primo livello e quelli regionali.
- I team IT delle unità aziendali potrebbero aver delegato il controllo dei rispettivi pool.
- I team applicativi potrebbero utilizzare gli indirizzi IP dei pool di ambienti designati.

> **Note**
>
> Puoi anche integrare questa soluzione con AWS Control Tower Account Factory for Terraform (AFT). Per ulteriori informazioni, vedere Integrazione con AFT nella sezione Informazioni aggiuntive di questo modello.

Tools (Strumenti)

Servizi AWS

- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue AWS risorse e delle applicazioni su cui esegui AWS in tempo reale.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire Servizi AWS tramite comandi nella shell della riga di comando.
- AWS Organizationsè un servizio di gestione degli account che consente di consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente.
- AWS Resource Access Manager (AWS RAM) ti aiuta a condividere in modo sicuro le tue risorse Account AWS per ridurre il sovraccarico operativo e fornire visibilità e verificabilità.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS. IP Address Manager è una funzionalità di Amazon VPC. Ti aiuta a pianificare, tracciare e monitorare gli indirizzi IP per i tuoi AWS carichi di lavoro.

Altri strumenti

- HashiCorp Terraform è uno strumento Infrastructure as Code (IaC) che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud.

Archivio di codici

Il codice per questo modello è disponibile nel Sample Terraform Implementation for Hierarchical IPAM sul repository on. AWS GitHub La struttura del repository include:

- Modulo root: orchestrazione della distribuzione e variabili di input.
- Modulo IPAM: implementazione principale dell'architettura descritta in questo modello.
- Modulo Tags: etichettatura standardizzata per tutte le risorse.

Best practice

Considerate le seguenti best practice per la pianificazione della rete:

- Pianifica prima: pianifica accuratamente lo spazio degli indirizzi IP prima dell'implementazione. Per ulteriori informazioni, consulta Piano per il provisioning degli indirizzi IP.
- Evita la sovrapposizione degli intervalli CIDR: assicurati che gli intervalli CIDR di ogni livello non si sovrappongano.
- Riserva spazio nel buffer: alloca sempre intervalli CIDR più ampi di quelli immediatamente necessari per far fronte alla crescita.
- Documenta l'allocazione degli indirizzi IP: conserva la documentazione della tua strategia di allocazione degli indirizzi IP.

Considerate le seguenti best practice di implementazione:

- Inizia con la non produzione: esegui prima l'implementazione in ambienti non di produzione.
- Usa la gestione dello stato di Terraform: implementa l'archiviazione e il blocco dello stato remoti. Per ulteriori informazioni, consulta Archiviazione e blocco dello stato nella documentazione di Terraform.
- Implementa il controllo della versione: controlla la versione di tutto il codice Terraform.
- CI/CD Implementazione dell'integrazione: utilizza pipeline di integrazione continua e distribuzione continua (CI/CD) per implementazioni ripetibili.

Considerate le seguenti best practice operative:

- Abilita l'importazione automatica: configura un pool di IP Address Manager per scoprire e importare automaticamente le risorse esistenti. Segui le istruzioni in Modifica un pool IPAM per attivare l'importazione automatica.
- Monitora l'utilizzo degli indirizzi IP: imposta allarmi per le soglie di utilizzo degli indirizzi IP. Per ulteriori informazioni, consulta Monitorare l'IPAM con Amazon CloudWatch.
- Verifica periodica: verifica periodicamente l'utilizzo e la conformità degli indirizzi IP. Per ulteriori informazioni, consulta Monitoraggio dell'utilizzo degli indirizzi IP in IPAM.
- Pulisci le allocazioni inutilizzate: rilascia le allocazioni degli indirizzi IP quando le risorse vengono disattivate. Per ulteriori informazioni, consulta Deprovisioning from a pool. CIDRs

Considerate le seguenti best practice di sicurezza:

- Implementa il privilegio minimo: utilizza i ruoli IAM con le autorizzazioni minime richieste. Per ulteriori informazioni, consulta le best practice di sicurezza in IAM e la gestione delle identità e degli accessi in IPAM.
- Utilizza le politiche di controllo dei servizi: implementa le politiche di controllo dei servizi (SCPs) per applicare l'utilizzo di IP Address Manager nella tua organizzazione. Per ulteriori informazioni, consulta Implementare l'uso di IPAM per la creazione di VPC con. SCPs
- Controlla la condivisione delle risorse: gestisci con attenzione l'ambito della condivisione delle risorse di IP Address Manager in. AWS RAM Per ulteriori informazioni, consulta Condividere un pool IPAM utilizzando AWS RAM.
- Applica i tag: implementa i tag obbligatori per tutte le risorse relative a IP Address Manager. Per ulteriori informazioni, consulta Strategia di tagging nella sezione Informazioni aggiuntive.

Epiche

Configura un account amministratore delegato per IP Address Manager

Operazione	Description	Competenze richieste
Abilita AWS Organizations le funzionalità.	Assicurati che tutte AWS Organizations le funzionalità siano abilitate. Per istruzioni, vedi Abilitazione di tutte	Amministratore AWS

Operazione	Description	Competenze richieste
	le funzionalità per un'organizzazione AWS Organizations nella AWS Organizations documentazione.	
Abilita la condivisione delle risorse in AWS RAM.	Utilizzando AWS CLI, inserisci il seguente comando per abilitare la condivisione AWS RAM delle risorse per la tua organizzazione: `aws ram enable-sharing-with-aws-organization` Per ulteriori informazioni, consulta Abilitare la condivisione delle risorse AWS Organizations nella AWS RAM documentazione.	Amministratore AWS

Operazione	Description	Competenze richieste
Designare un amministratore per IP Address Manager.	Dall'account di gestione dell'organizzazione, utilizzando AWS CLI, immettete il seguente comando, `123456789012` dov'è l'ID dell'account che amministrerà IP Address Manager: `aws ec2 enable-ipam-organization-admin-account \ --delegated-admin-account-id 123456789012` **Note** In genere, un account di rete o di hub di rete viene utilizzato come amministratore delegato per IP Address Manager. Per ulteriori informazioni, vedere Integrazione di IPAM con gli account di un' AWS organizzazione nella documentazione di IP Address Manager.	Amministratore AWS

Implementa l'infrastruttura

Operazione	Description	Competenze richieste
Definire l'architettura di rete.	Definite e documentate l'architettura di rete, compresi gli intervalli CIDR per regioni, unità aziendali e ambienti. Per ulteriori informazioni, consulta Piano per il provisioning degli indirizzi IP nella documenta zione di IP Address Manager.	Ingegnere di rete
Clonare il repository.	1. Immettete il seguente comando per clonare il repository sulla vostra workstation locale: `git clone https://github.com/aws-samples/sample-amazon-vpc-ipam-terraform.git` 2. Immettete il seguente comando per navigare nella directory. `terraform`	DevOps ingegnere
Configura le variabili.	1. Apri il file terraform.tfvars. 2. Configura le variabili per la distribuzione di IP Address Manager. Per un esempio di file completato, consulta terraform.tfvars.example nel repository del codice. 3. Salva e chiudi il file terraform.tfvars.	Ingegnere di rete, Terraform

Operazione	Description	Competenze richieste

Implementa le risorse di IP Address Manager.	1. Inserisci il seguente comando per inizializzare Terraform:
	`terraform init`
	2. Inserisci il seguente comando per generare un piano di esecuzione:
	`terraform plan`
	3. Rivedi il piano e convalida le risorse e i componenti dell'infrastruttura che verranno creati.
	4. Immettere il seguente comando per distribuire l'infrastruttura:
	`terraform apply`
	5. Quando richiesto, inserisci yes per confermare la distribuzione.
	6. Attendi il completamento della distribuzione.
		Terraform

Operazione	Description	Competenze richieste
Convalida la distribuzione.	1. Accedi al tuo Account AWS, quindi apri la console di IP Address Manager. 2. Nel pannello di navigazione, seleziona Pool. 3. Nell'elenco dei pool, verifica che i pool configurati siano stati creati. 4. Apri la AWS RAM console. 5. Verifica che le risorse di IP Address Manager siano state condivise. Segui le istruzioni riportate nella sezione Visualizzazione delle condivisioni di risorse che hai creato nella AWS RAM documentazione.	General AWS, ingegnere di rete

Crea VPCs e configura il monitoraggio

Operazione	Description	Competenze richieste
Crea un VPC.	Segui i passaggi in Creare un VPC nella documentazione di Amazon VPC. Quando raggiungi la fase di scelta di un intervallo CIDR per il VPC, alloca il successivo disponibile da uno dei tuoi pool regionali, aziendali e ambientali.	General AWS, amministratore di rete, ingegnere di rete
Convalida l'allocazione dell'intervallo CIDR.	1. Aprire la console di IP Address Manager.	General AWS, amministratore di rete, ingegnere di rete

Operazione	Description	Competenze richieste
	2. Nel pannello di navigazione seleziona Pannello di controllo. 3. Nella dashboard, verifica che l'intervallo CIDR sia stato allocato al VPC che hai creato.	
Monitora il gestore degli indirizzi IP.	Configura il monitoraggio e gli allarmi relativi all'allocazione delle risorse di IP Address Manager. Per ulteriori informazioni e istruzioni, consulta Monitoraggio dell'IPAM con Amazon CloudWatch e Monitoraggio dell'utilizzo del CIDR per risorsa nella documentazione di IP Address Manager.	Informazioni generali su AWS
Imponi l'uso di IP Address Manager.	Crea una policy di controllo dei servizi (SCP) AWS Organizations che richieda ai membri dell'organizzazione di utilizzare IP Address Manager quando creano un VPC. Per istruzioni, consulta Implementare l'uso di IPAM per la creazione di VPC SCPs nella documentazione di IP Address Manager.	AWS generale, amministratore AWS

risoluzione dei problemi

Problema	Soluzione
Terraform fallisce con la risorsa IP Address Manager non trovata	Assicurati che l'account amministratore di IP Address Manager sia delegato correttamente e che il tuo AWS provider sia autenticato su quell'account.
L'allocazione dell'intervallo CIDR non riesce	Verifica che l'intervallo CIDR richiesto rientri nell'intervallo disponibile del pool di IP Address Manager e non si sovrapponga alle allocazioni esistenti.
AWS RAM problemi di condivisione	Verifica che la condivisione delle risorse sia abilitata per la tua AWS organizzazione. Verifica che nella condivisione venga utilizzato il principale corretto, l'organizzazione Amazon Resource Name (ARN). AWS RAM
Errori di convalida della gerarchia del pool	Assicurati che gli intervalli CIDR del pool secondario siano contenuti correttamente negli intervalli CIDR del pool principale e non si sovrappongano ai pool di pari livello.
Il limite di quota di IP Address Manager è stato superato	Richiedere un aumento della quota per i pool di IP Address Manager. Per ulteriori informazioni, consulta Richiesta di un aumento delle quote nella Guida per l'utente di Service Quotas.

Risorse correlate

Servizio AWS documentazione

- Documentazione di Amazon VPC IP Address Manager
- AWS Resource Access Manager documentazione
- AWS Organizations documentazione

AWS post sul blog

- Gestione di pool di IP in tutte VPCs le regioni utilizzando Amazon VPC IP Address Manager
- Gestione e controllo degli indirizzi di rete su larga scala con Amazon VPC IP Address Manager

Video e tutorial

- AWS re:Invent 2022: best practice per la progettazione di Amazon VPC e IPAM (0) NET31
- AWS re:Invent 2022: progettazione VPC avanzata e nuove funzionalità (01) NET4

Informazioni aggiuntive

Integrazione con AFT

Puoi integrare questa soluzione con AWS Control Tower Account Factory for Terraform (AFT) per assicurarti che gli account appena assegnati ricevano automaticamente le configurazioni di rete corrette. Implementando questa soluzione IPAM nell'account dell'hub di rete, i nuovi account creati tramite AFT possono fare riferimento ai pool di IP Address Manager condivisi durante la creazione. VPCs

Il seguente esempio di codice dimostra l'integrazione AFT nella personalizzazione di un account utilizzando Parameter Store: AWS Systems Manager

```
# Get the IP Address Manager pool ID from Parameter Store
data "aws_ssm_parameter" "dev_ipam_pool_id" {
  name = "/org/network/ipam/finance/dev/pool-id"
}

# Create a VPC using the IP Address Manager pool
resource "aws_vpc" "this" {
  ipv4_ipam_pool_id   = data.aws_ssm_parameter.dev_ipam_pool_id.value
  ipv4_netmask_length = 24

  tags = {
    Name = "aft-account-vpc"
  }
}
```

Strategia di assegnazione tag

La soluzione implementa una strategia di tagging completa per facilitare la gestione delle risorse. Il seguente esempio di codice ne illustra l'utilizzo:

```
# Example tag configuration
module "tags" {
  source = "./modules/tags"

  # Required tags
  product_name  = "enterprise-network"
  feature_name  = "ipam"
  org_id        = "finance"
  business_unit = "network-operations"
  owner         = "network-team"
  environment   = "prod"
  repo          = "https://github.com/myorg/ipam-terraform"
  branch        = "main"
  cost_center   = "123456"
  dr_tier       = "tier1"

  # Optional tags
  optional_tags = {
    "project"    = "network-modernization"
    "stack_role" = "infrastructure"
  }
}
```

Questi tag vengono applicati automaticamente a tutte le risorse di IP Address Manager. Ciò facilita la governance, l'allocazione dei costi e la gestione delle risorse coerenti.

Personalizza CloudWatch gli avvisi Amazon per AWS Network Firewall

Jason Owens, Amazon Web Services

Riepilogo

Il modello ti aiuta a personalizzare gli CloudWatch avvisi Amazon generati da AWS Network Firewall. Puoi utilizzare regole predefinite o creare regole personalizzate che determinano il messaggio, i metadati e la gravità degli avvisi. Puoi quindi agire in base a questi avvisi o automatizzare le risposte di altri servizi Amazon, come Amazon. EventBridge

In questo modello, si generano regole firewall compatibili con Suricata. Suricata è un motore di rilevamento delle minacce open source. Per prima cosa devi creare regole semplici e poi testarle per confermare che CloudWatch gli avvisi vengano generati e registrati. Dopo aver testato con successo le regole, le modifichi per definire messaggi, metadati e livelli di severità personalizzati, quindi esegui nuovamente il test per confermare gli aggiornamenti.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- AWS Command Line Interface (AWS CLI) installato e configurato sulla tua workstation Linux, macOS o Windows. Per ulteriori informazioni, consulta Installare o aggiornare la versione più recente della AWS CLI.
- AWS Network Firewall installato e configurato per utilizzare CloudWatch Logs. Per ulteriori informazioni, vedere Registrazione del traffico di rete da. AWS Network Firewall
- Un'istanza Amazon Elastic Compute Cloud (Amazon EC2) in una sottorete privata di un cloud privato virtuale (VPC) protetto da Network Firewall.

Versioni del prodotto

- Per la versione 1 di AWS CLI, usa 1.18.180 o successiva. Per la versione 2 di AWS CLI, usa 2.1.2 o successiva.
- Il file classification.config della versione 5.0.2 di Suricata. Per una copia di questo file di configurazione, vedere la sezione Informazioni aggiuntive.

Architecture

Il diagramma dell'architettura mostra il seguente flusso di lavoro:

1. Un' EC2 istanza Amazon in una sottorete privata effettua una richiesta utilizzando curl o Wget.
2. Network Firewall elabora il traffico e genera un avviso.
3. Network Firewall invia gli avvisi registrati ai CloudWatch registri.

Tools (Strumenti)

Servizi AWS

- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue AWS risorse e delle applicazioni su cui esegui AWS in tempo reale.
- Amazon CloudWatch Logs ti aiuta a centralizzare i log di tutti i tuoi sistemi e applicazioni, Servizi AWS così puoi monitorarli e archiviarli in modo sicuro.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire Servizi AWS tramite comandi nella shell della riga di comando.
- AWS Network Firewallè un firewall di rete a stato gestito e un servizio di rilevamento e prevenzione delle intrusioni per cloud privati virtuali () in. VPCs Cloud AWS

Altri strumenti

- curl è uno strumento e una libreria a riga di comando open source.
- GNU Wget è uno strumento da riga di comando gratuito.

Epiche

Crea le regole del firewall e il gruppo di regole

Operazione	Description	Competenze richieste
Creare regole.	1. In un editor di testo, create un elenco di regole da aggiungere al firewall. Ogni	Amministratore di sistema AWS, amministratore di rete

Operazione	Description	Competenze richieste

	regola deve trovarsi su una riga distinta. Il valore del `classtype` parametro proviene dal file di configurazione di classificazione Suricata predefinito. Per il contenuto completo del file di configurazione, vedere la sezione Informazioni aggiuntive. Di seguito sono riportati due esempi di regole.
	`alert http any any -> any any (content:"badstuff"; classtype:misc-activity; sid:3; rev:1;)`
	`alert http any any -> any any (content:"morebadstuff"; classtype:bad-unknown; sid:4; rev:1;)`
	2. Salva le regole in un file denominato`custom.rules` .	

Operazione	Description	Competenze richieste

Crea il gruppo di regole.	Nel AWS CLI, immettere il seguente comando. Questo crea il gruppo di regole.
	
	`# aws network-firewall create-rule-group \ --rule-group-name custom --type STATEFUL \ --capacity 10 --rules file://custom.rules \ --tags Key=environment,Value=development`
	
	Di seguito è riportato un esempio di output. Prendi nota di`RuleGroupArn` , che ti servirà in un passaggio successivo.
	
	`{ "UpdateToken": "4f998d72-973c-490a-bed2-fc3460547e23", "RuleGroupResponse": { "RuleGroupArn": "arn:aws:network-firewall:us-east-2:1234567890:stateful-rulegroup/custom", "RuleGroupName": "custom", "RuleGroupId": "238a8259-9eaf-48bb-90af-5e690cf8c48b",`
		Amministratore di sistema AWS

Operazione	Description	Competenze richieste

	<pre>"Type": "STATEFUL",
	"Capacity": 10,
	"RuleGroupStatus": "ACTIVE",
	"Tags": [
	{
	"Key": "environment",
	"Value": "development"
	}
	]
	}</pre>	

Aggiorna la politica del firewall

Operazione	Description	Competenze richieste

Ottieni l'ARN della politica del firewall.	Nel AWS CLI, immettere il seguente comando. Ciò restituisce l'Amazon Resource Name (ARN) della policy del firewall. Registra l'ARN per utilizzarlo più avanti in questo schema.
	<pre># aws network-firewall describe-firewall \
	--firewall-name aws-network-firewall-anfw \
	--query 'Firewall.FirewallPolicyArn'</pre>
		Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	Di seguito è riportato un esempio di ARN restituito da questo comando. `"arn:aws:network-firewall:us-east-2:1234567890:firewall-policy/firewall-policy-anfw"`	

Operazione	Description	Competenze richieste

Aggiorna la politica del firewall.	In un editor di testo, copia o incolla il seguente codice. Sostituiscilo `<RuleGroupArn>` con il valore che hai registrato nell'epopea precedente. Salva il file con nome `firewall-policy-anfw.json` .
	
	`{`
	`  "StatelessDefaultActions": [`
	`  "aws:forward_to_sfe"`
	`  ],`
	`  "StatelessFragmentDefaultActions": [`
	`  "aws:forward_to_sfe"`
	`  ],`
	`  "StatefulRuleGroupReferences": [`
	`  {`
	`  "ResourceArn": "<RuleGroupArn>"`
	`  }`
	`  ]`
	`}`
	
	Inserisci il seguente comando in. AWS CLI Questo comando richiede un token di aggiornamento per aggiungere le nuove regole. Il token viene utilizzato per confermare che la politica non è cambiata dall'ultima volta che l'hai recuperata.
		Amministratore di sistema AWS

Operazione	Description	Competenze richieste

	`UPDATETOKEN=(`aws network-firewall describe-firewall-policy \ --firewall-policy-name firewall-policy-anfw \ --output text --query UpdateToken`)`
	
	`aws network-firewall update-firewall-policy \ --update-token $UPDATETOKEN \ --firewall-policy-name firewall-policy-anfw \ --firewall-policy file://firewall-policy-anfw.json`	

Operazione	Description	Competenze richieste

Conferma gli aggiornamenti delle policy.	(Facoltativo) Se desideri confermare che le regole sono state aggiunte e visualizzare il formato della politica, inserisci il seguente comando in AWS CLI.
	
	`# aws network-firewall describe-firewall-policy \ --firewall-policy-name firewall-policy-anfw \ --query FirewallPolicy`
	
	Di seguito è riportato un esempio di output.
	
	`{ "StatelessDefaultActions": [ "aws:forward_to_sfe" ], "StatelessFragmentDefaultActions": [ "aws:forward_to_sfe" ], "StatefulRuleGroupReferences": [ { "ResourceArn": "arn:aws:network-firewall:us-east-2:1234567890:stateful-rulegroup/custom"`
		Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	<pre> }] }</pre>	

Verifica la funzionalità degli avvisi

Operazione	Description	Competenze richieste

Genera avvisi per i test.	1. Accedi a una workstation di prova all'interno della sottorete del firewall.
	2. Immettete i comandi che dovrebbero generare avvisi. Ad esempio, puoi usare `wget` o `curl`.
	<pre>wget -U "badstuff"
	 http://www.amazon.
	com -o /dev/null</pre>
	<pre>curl -A "morebads
	tuff" http://ww
	w.amazon.com -o /
	dev/null</pre>
		Amministratore di sistema AWS

Verifica che gli avvisi siano registrati.	1. Apri la CloudWatch console. 2. Passa al gruppo di log e allo stream corretti. Per ulteriori informazioni, consulta Visualizza i dati di registro inviati ai CloudWatch registri (documentazione CloudWatch sui registri).	Amministratore di sistema AWS

Operazione	Description	Competenze richieste

	3. Verifica che gli eventi registrati siano simili agli esempi seguenti. Gli esempi mostrano solo la parte rilevante dell'avviso.
	
	Esempio 1
	
	`"alert": { "action": "allowed", "signature_id": 3, "rev": 1, "signature": "", "category": "Misc activity", "severity": 3 }`
	
	Esempio 2
	
	`"alert": { "action": "allowed", "signature_id": 4, "rev": 1, "signature": "", "category": "Potentially Bad Traffic", "severity": 2 }`	

Aggiorna le regole e il gruppo di regole del firewall

Operazione	Description	Competenze richieste
Aggiorna le regole del firewall.	1. In un editor di testo, aprire il file `custom.rules` . 2. Modificate la prima regola in modo che sia simile alla seguente. Questa regola deve essere inserita su una sola riga del file. `alert http any any -> any any (msg:"Watch out - Bad Stuff!!"; content:"badstuff"; classtype:misc-activity; priority:2; sid:3; rev:2; metadata:custom-field-2 Danger!, custom-field More Info;)` Ciò apporta le seguenti modifiche alla regola: • Aggiunge una stringa msg (sito Web Suricata) che fornisce informazioni testuali sulla firma o sull'avviso. Nell'avviso generato, questo corrisponde alla firma. • Regola la priorità predefinita (sito Web Suricata) `misc-activity` da 3 a 2. Per i valori predefiniti dei	Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	varclasstypes , vedere la sezione Informazioni aggiuntive. • Aggiunge metadati personalizzati (sito Web Suricata) all'avviso. Si tratta di informazioni aggiuntive che vengono aggiunte alla firma. Si consiglia di utilizzare coppie chiave-valore. • Cambia la versione (sito web di Suricata) da 1 a 2. Rappresenta la versione della firma.	

Operazione	Description	Competenze richieste

Aggiorna il gruppo di regole.	In AWS CLI, esegui i seguenti comandi. Usa l'ARN della tua politica firewall. Questi comandi ottengono un token di aggiornamento e aggiornano il gruppo di regole con le modifiche alle regole.
	`# UPDATETOKEN=(`aws network-firewall \ describe-rule-group \ --rule-group-arn arn:aws:network-firewall:us-east-2:123457890:stateful-rulegroup/custom \ --output text --query UpdateToken`)`
	`# aws network-firewall update-rule-group \ --rule-group-arn arn:aws:network-firewall:us-east-2:1234567890:stateful-rulegroup/custom \ --rules file://custom.rules \ --update-token $UPDATETOKEN`
	Di seguito è riportato un esempio di output.
	`{`
		Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	"UpdateToken": "7536939f-6a1d-414c-96d1-bb28110996ed", "RuleGroupResponse": { "RuleGroupArn": "arn:aws:network-firewall:us-east-2:1234567890:stateful-rulegroup/custom", "RuleGroupName": "custom", "RuleGroupId": "238a8259-9eaf-48bb-90af-5e690cf8c48b", "Type": "STATEFUL", "Capacity": 10, "RuleGroupStatus": "ACTIVE", "Tags": [{ "Key": "environment", "Value": "development" }] } }	

Prova la funzionalità di avviso aggiornata

Operazione	Description	Competenze richieste
Genera un avviso per il test.	1. Accedere a una workstation di prova all'interno della sottorete del firewall.	Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	2. Immettete un comando che dovrebbe generare un avviso. Per esempio, è possibile utilizzare `curl`. `curl -A "badstuff" http://www.amazon.com -o /dev/null`	

Operazione	Description	Competenze richieste
Convalida l'avviso modificato.	1. Apri la CloudWatch console. 2. Passa al gruppo di log e allo stream corretti. 3. Conferma che l'evento registrato è simile all'esempio seguente. L'esempio mostra solo la parte rilevante dell'avviso. `"alert": { "action": "allowed", "signature_id": 3, "rev": 2, "signature": "Watch out - Bad Stuff!!", "category": "Misc activity", "severity": 2, "metadata": { "custom-field": [ "More Info" ], "custom-field-2": [ "Danger!" ] } }`	Amministratore di sistema AWS

Risorse correlate

Riferimenti

- Invia avvisi da AWS Network Firewall un canale Slack (AWS Prescriptive Guidance)
- Proseguire la prevenzione delle minacce AWS con Suricata (post sul blog)AWS
- Modelli di implementazione per AWS Network Firewall (AWS post sul blog)
- Suricata meta keyworks (documentazione Suricata)

Tutorial e video

- AWS Network Firewall officina

Informazioni aggiuntive

Di seguito è riportato il file di configurazione della classificazione di Suricata 5.0.2. Queste classificazioni vengono utilizzate durante la creazione delle regole del firewall.

```
# config classification:shortname,short description,priority

config classification: not-suspicious,Not Suspicious Traffic,3
config classification: unknown,Unknown Traffic,3
config classification: bad-unknown,Potentially Bad Traffic, 2
config classification: attempted-recon,Attempted Information Leak,2
config classification: successful-recon-limited,Information Leak,2
config classification: successful-recon-largescale,Large Scale Information Leak,2
config classification: attempted-dos,Attempted Denial of Service,2
config classification: successful-dos,Denial of Service,2
config classification: attempted-user,Attempted User Privilege Gain,1
config classification: unsuccessful-user,Unsuccessful User Privilege Gain,1
config classification: successful-user,Successful User Privilege Gain,1
config classification: attempted-admin,Attempted Administrator Privilege Gain,1
config classification: successful-admin,Successful Administrator Privilege Gain,1

# NEW CLASSIFICATIONS
config classification: rpc-portmap-decode,Decode of an RPC Query,2
config classification: shellcode-detect,Executable code was detected,1
config classification: string-detect,A suspicious string was detected,3
config classification: suspicious-filename-detect,A suspicious filename was detected,2
```

```
config classification: suspicious-login,An attempted login using a suspicious username
 was detected,2
config classification: system-call-detect,A system call was detected,2
config classification: tcp-connection,A TCP connection was detected,4
config classification: trojan-activity,A Network Trojan was detected, 1
config classification: unusual-client-port-connection,A client was using an unusual
 port,2
config classification: network-scan,Detection of a Network Scan,3
config classification: denial-of-service,Detection of a Denial of Service Attack,2
config classification: non-standard-protocol,Detection of a non-standard protocol or
 event,2
config classification: protocol-command-decode,Generic Protocol Command Decode,3
config classification: web-application-activity,access to a potentially vulnerable web
 application,2
config classification: web-application-attack,Web Application Attack,1
config classification: misc-activity,Misc activity,3
config classification: misc-attack,Misc Attack,2
config classification: icmp-event,Generic ICMP event,3
config classification: inappropriate-content,Inappropriate Content was Detected,1
config classification: policy-violation,Potential Corporate Privacy Violation,1
config classification: default-login-attempt,Attempt to login by a default username and
 password,2

# Update
config classification: targeted-activity,Targeted Malicious Activity was Detected,1
config classification: exploit-kit,Exploit Kit Activity Detected,1
config classification: external-ip-check,Device Retrieving External IP Address
 Detected,2
config classification: domain-c2,Domain Observed Used for C2 Detected,1
config classification: pup-activity,Possibly Unwanted Program Detected,2
config classification: credential-theft,Successful Credential Theft Detected,1
config classification: social-engineering,Possible Social Engineering Attempted,2
config classification: coin-mining,Crypto Currency Mining Activity Detected,2
config classification: command-and-control,Malware Command and Control Activity
 Detected,1
```

Distribuisci risorse in una AWS Wavelength zona utilizzando Terraform

Zahoor Chaudhrey e Luca Iannario, Amazon Web Services

Riepilogo

AWS Wavelengthti aiuta a creare un'infrastruttura ottimizzata per le applicazioni Multi-Access Edge Computing (MEC). Le Wavelength Zones AWS sono implementazioni di infrastrutture che AWS incorporano servizi di elaborazione e archiviazione nelle reti 5G dei provider di servizi di comunicazione (CSP). Il traffico delle applicazioni proveniente dai dispositivi 5G raggiunge i server delle applicazioni in esecuzione nelle Wavelength Zones senza uscire dalla rete di telecomunicazioni. Quanto segue facilita la connettività di rete tramite Wavelength:

- Cloud privati virtuali (VPCs): VPCs Account AWS possono estendersi fino a coprire più zone di disponibilità, incluse le Wavelength Zone. Le istanze di Amazon Elastic Compute Cloud (Amazon EC2) e i servizi correlati vengono visualizzati come parte del tuo VPC regionale. VPCs vengono creati e gestiti in Amazon Virtual Private Cloud (Amazon VPC).
- Carrier gateway: un carrier gateway consente la connettività dalla sottorete nella Wavelength Zone alla rete CSP, a Internet o tramite la rete Regione AWS del CSP. Il carrier gateway serve a due scopi. Consente il traffico in entrata da una rete CSP in una posizione specifica e consente il traffico in uscita verso la rete di telecomunicazioni e Internet.

Questo pattern e il codice Terraform associato ti aiutano ad avviare risorse, come EC2 istanze Amazon, volumi Amazon Elastic Block Store (Amazon EBS), sottoreti e un carrier gateway VPCs, in una Wavelength Zone.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un ambiente di sviluppo integrato (IDE)
- Attiva la Wavelength Zone di destinazione
- AWS Command Line Interface (AWS CLI), installato e configurato
- Terraform versione 1.8.4 o successiva, installata (documentazione Terraform)
- Terraform AWS Provider versione 5.32.1 o successiva, configurata (documentazione Terraform)
- Git, installato (GitHub)

- Autorizzazioni per creare risorse Amazon VPC, Wavelength e Amazon EC2

Limitazioni

Non tutti Regioni AWS supportano le Wavelength Zones. Per ulteriori informazioni, consulta Available Wavelength Zones nella documentazione di Wavelength.

Architecture

Il diagramma seguente mostra come creare una sottorete e AWS risorse in una Wavelength Zone. VPCs che contengono una sottorete in una Wavelength Zone possono connettersi a un gateway carrier. Un carrier gateway consente di connettersi alle seguenti risorse:

- Dispositivi 4G/LTE e 5G sulla rete del gestore di telecomunicazioni.
- Accesso wireless fisso per partner Wavelength Zone selezionati. Per ulteriori informazioni, vedere Accesso multiplo. AWS Wavelength
- Traffico in uscita verso risorse Internet pubbliche.

Tools (Strumenti)

Servizi AWS

- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.
- AWS Wavelengthestende Cloud AWS l'infrastruttura alle reti 5G dei provider di telecomunicazioni. Questo ti aiuta a creare applicazioni che offrono latenze estremamente basse ai dispositivi mobili e agli utenti finali.

Altri strumenti

- Terraform è uno strumento di infrastruttura come codice (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.

Archivio di codici

Il codice per questo modello è disponibile nel repository GitHub Creating AWS Wavelength Infrastructure using Terraform. Il codice Terraform implementa la seguente infrastruttura e risorse:

- Un VPC
- Una zona di Wavelength
- Una sottorete pubica nella Wavelength Zone
- Un carrier gateway nella Wavelength Zone
- Un' EC2 istanza Amazon nella Wavelength Zone

Best practice

- Prima della distribuzione, conferma di utilizzare le versioni più recenti di Terraform e di. AWS CLI
- Utilizza una pipeline di integrazione e distribuzione continua (CI/CD) per implementare IaC. Per ulteriori informazioni, consulta Best practice per la gestione dei file Terraform State in CI/CD Pipeline on Blogs. AWS AWS

Epiche

Fornisci l'infrastruttura

Operazione	Description	Competenze richieste
Clonare il repository.	Inserisci il seguente comando per clonare l'AWS Wavelength infrastruttura di creazione utilizzando il repository Terraform nel tuo ambiente. `git clone git@github.com:aws-samples/terraform-wavelength-infrastructure.git`	DevOps ingegnere
Aggiorna le variabili.	1. Vai al repository clonato.	DevOps ingegnere, Terraform

Operazione	Description	Competenze richieste
	`cd terraform-waveleng th-infrastructure` 2. Usando qualsiasi editor di testo, crea un file chiamato terraform.tfvars nella directory principale. 3. Create le seguenti variabili e inserite i relativi valori: • `region = <enter Region name>` • `vpc_cidr = <enter CIDR block used by VPC>` • `wavelength_subnet_ cidr = <enter CIDR block for the subnet in the Wavelength Zone>` • `availabilityzone_w avelength = <enter Wavelength Zone name>` 4. Salva il file terraform.tfvars.	
Inizializza la configurazione.	Immettete il seguente comando per inizializzare la directory di lavoro. `terraform init`	DevOps ingegnere, Terraform

Operazione	Description	Competenze richieste
Visualizza l'anteprima del piano Terraform.	Immettete il seguente comando per confrontare lo stato di destinazione con lo stato attuale del vostro AWS ambiente. Questo comando genera un'anteprima delle risorse che verranno configurate. `terraform plan`	DevOps ingegnere, Terraform
Verifica e distribuisci.	1. Rivedi le modifiche alla configurazione nel piano Terraform e conferma di voler implementare queste modifiche. 2. Immettere il seguente comando per applicare il piano e creare l'infrastruttura. `terraform apply` 3. Entra `yes` per procedere. Terraform crea l'architettura dichiarata nei file di configurazione. Per ulteriori informazioni sull'architettura, consulta la sezione Architettura di Target di questo modello.	DevOps ingegnere, Terraform

Convalida e ripulisci

Operazione	Description	Competenze richieste
Verifica l'implementazione dell'infrastruttura.	1. Se non hai già un' EC2 istanza Amazon in una sottorete pubblica Regione AWS, creane una. Per istruzioni, consulta Avvio dell'istanza Linux o Avvio dell'istanza Windows. Utilizzerai questa istanza per testare la connettività dalla Wavelength Zone Regione AWS alla Wavelength Zone. 2. Verifica la connettività dall'istanza Regione AWS all'istanza nella Wavelength Zone. Per istruzioni, consulta Verifica della connettività nella documentazione di Wavelength.	AWS DevOps, DevOps ingegnere
(Facoltativo) Pulisci l'infrastruttura.	Se devi eliminare tutte le risorse fornite da Terraform, procedi come segue: 1. Inserire il seguente comando. `terraform destroy` 2. Immettere `yes` per confermare.	DevOps ingegnere, Terraform

risoluzione dei problemi

Problema	Soluzione
Connettività alle EC2 istanze Amazon in. Regione AWS	Vedi Risoluzione dei problemi di connessione alla tua istanza Linux o Risoluzione dei problemi di connessione alla tua istanza Windows.
Connettività alle EC2 istanze Amazon nella Wavelength Zone.	Vedi Risoluzione dei problemi di connettività SSH o RDP per le mie EC2 istanze avviate in una Wavelength Zone.
Capacità nella Wavelength Zone.	Vedi Quote e considerazioni per Wavelength Zones.
Connettività mobile o gestionale dalla rete del gestore a. Regione AWS	1. Verifica che il gateway dell'operatore sia operativo. Esegui questa operazione: a. Apri la Console Amazon VPC. b. Nel pannello di navigazione, scegli Your VPCs. c. Seleziona il VPC che contiene la Wavelength Zone. d. Nel riquadro Dettagli, per Carrier gateway, conferma che il valore sia allegato. 2. Verifica che tutti gli indirizzi IP elastici collegati alle istanze nella Wavelength Zone siano operativi. Esegui questa operazione: a. Apri la EC2 console Amazon. b. Nel riquadro di navigazione, scegliere Instances (Istanze). c. Selezionate l'istanza nella Wavelength Zone. d. Scegli la scheda Rete.

Problema	Soluzione
	e. Verifica che all'elastic network interface sia associato un indirizzo IP elastico. 3. Contatta il team di supporto della rete dell'operatore.

Risorse correlate

- Che cos'è AWS Wavelength?
- Come AWS Wavelength funziona
- Resilienza in AWS Wavelength

Esegui la migrazione di record DNS in blocco verso una zona ospitata privata di Amazon Route 53

Ram Kandaswamy, Amazon Web Services

Riepilogo

Gli ingegneri di rete e gli amministratori del cloud hanno bisogno di un modo semplice ed efficiente per aggiungere record DNS (Domain Name System) alle zone private ospitate in Amazon Route 53. L'utilizzo di un approccio manuale per copiare le voci da un foglio di lavoro di Microsoft Excel nelle posizioni appropriate nella console Route 53 è noioso e soggetto a errori. Questo modello descrive un approccio automatizzato che riduce il tempo e lo sforzo necessari per aggiungere più record. Fornisce inoltre una serie di passaggi ripetibili per la creazione di più zone ospitate.

Questo modello utilizza Amazon Simple Storage Service (Amazon S3) per archiviare i record. Per lavorare con i dati in modo efficiente, il pattern utilizza il formato JSON per la sua semplicità e la sua capacità di supportare un dizionario Python `dict` (tipo di dati).

> **Note**
>
> Se riesci a generare un file di zona dal tuo sistema, prendi in considerazione l'utilizzo della funzione di importazione Route 53.

Prerequisiti e limitazioni

Prerequisiti

- Un foglio di lavoro Excel che contiene i record delle zone ospitate private
- Familiarità con diversi tipi di record DNS come A record, Name Authority Pointer (NAPTR) e record SRV (vedi Tipi di record DNS supportati)
- Familiarità con il linguaggio Python e le sue librerie

Limitazioni

- Il modello non fornisce una copertura estesa per tutti gli scenari di utilizzo. Ad esempio, la chiamata change_resource_record_sets non utilizza tutte le proprietà disponibili dell'API.

- Nel foglio di lavoro di Excel, si presume che il valore di ogni riga sia univoco. È previsto che nella stessa riga compaiano più valori per ogni nome di dominio completo (FQDN). Se ciò non è vero, è necessario modificare il codice fornito in questo modello per eseguire la concatenazione necessaria.
- Il modello utilizza l'SDK AWS per Python (Boto3) per chiamare direttamente il servizio Route 53. Puoi migliorare il codice per utilizzare un CloudFormation wrapper AWS per i `update_stack` comandi `create_stack` and e utilizzare i valori JSON per popolare le risorse del modello.

Architecture

Stack tecnologico

- Zone ospitate private Route 53 per il routing del traffico
- Amazon S3 per l'archiviazione del file JSON di output

Il flusso di lavoro consiste nei seguenti passaggi, come illustrato nel diagramma precedente e discusso nella sezione Epics:

1. Carica un foglio di lavoro Excel contenente le informazioni sul set di record in un bucket S3.
2. Crea ed esegui uno script Python che converta i dati di Excel in formato JSON.
3. Leggi i record dal bucket S3 e pulisci i dati.
4. Crea set di record nella tua zona ospitata privata.

Tools (Strumenti)

- Route 53 — Amazon Route 53 è un servizio Web DNS altamente disponibile e scalabile che gestisce la registrazione del dominio, il routing DNS e il controllo dello stato.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti. È possibile utilizzare Amazon S3 per memorizzare e recuperare qualsiasi volume di dati, in qualunque momento e da qualunque luogo tramite il Web.

Epiche

Prepara i dati per l'automazione

Operazione	Description	Competenze richieste
Crea un file Excel per i tuoi archivi.	Usa i record che hai esportato dal sistema corrente per creare un foglio di lavoro Excel contenente le colonne richieste per un record, ad esempio nome di dominio completo (FQDN), tipo di record, Time to Live (TTL) e valore. Per i record NAPTR e SRV, il valore è una combinazione di più proprietà, quindi usa il metodo di Excel per combinare queste proprietà. `concat` FqdnN Recor e Valore TTL qualcc exam rg A 1.1.1.1 900	Ingegnere dei dati, competenze in Excel
Verifica l'ambiente di lavoro.	Nel tuo IDE, crea un file Python per convertire il foglio di lavoro di input di Excel in formato JSON. (Invece di un IDE, puoi anche usare un SageMaker notebook Amazon per lavorare con il codice Python.)	Informazioni generali su AWS

Operazione	Description	Competenze richieste

	Verifica che la versione di Python che stai utilizzando sia la versione 3.7 o successiva.
	`python3 --version`
	Installa il pacchetto pandas.
	`pip3 install pandas --user`	

Converti i dati del foglio di lavoro Excel in JSON.	Crea un file Python che contenga il seguente codice da convertire da Excel a JSON.
	`import pandas as pd`
	`data=pd.read_excel('./Book1.xls')`
	`data.to_json(path_or_buf='my.json', orient='records')`
	dove `Book1` è il nome del foglio di lavoro di Excel ed `my.json` è il nome del file JSON di output.
		Ingegnere dei dati, competenze in Python

Carica il file JSON in un bucket S3.	Caricare il file `my.json` in un bucket S3. Per ulteriori informazioni, consulta Creazione di un bucket nella documentazione di Amazon S3.	Sviluppatore di app

Inserisci record

Operazione	Description	Competenze richieste

Crea una zona ospitata privata.	Usa l'API create_hosted_zone e il seguente codice di esempio Python per creare una zona ospitata privata. Sostituisci i parametri e con i tuoi valori `hostedZoneName` . `vpcRegion` `vpcId`
	`import boto3`
	`import random`
	`hostedZoneName ="xxx"`
	`vpcRegion = "us-east-1"`
	`vpcId="vpc-xxxx"`
	`route53_client = boto3.client('route53')`
	`response = route53_client.create_hosted_zone(`
	`Name= hostedZoneName,`
	`VPC={`
	`'VPCRegion: vpcRegion,`
	`'VPCId': vpcId`
	`},`
	`CallerReference=str(random.random()*100000),`
	`HostedZoneConfig={`
	`'Comment': "private hosted zone created by automation",`
		Architetto cloud, amministratore di rete, competenze in Python

Operazione	Description	Competenze richieste

	<pre>'PrivateZone': True
	}
	)
	print(response)</pre>
	Puoi anche utilizzare uno strumento di infrastruttura come codice (IaC) come AWS CloudFormation per sostituire questi passaggi con un modello che crea uno stack con le risorse e le proprietà appropriate.	

Recupera i dettagli come dizionario da Amazon S3.	Usa il codice seguente per leggere dal bucket S3 e ottenere i valori JSON come dizionario Python.
	<pre>fileobj = s3_client.get_object(
	Bucket=bucket_name,
	Key='my.json'
	)
	filedata = fileobj['Body'].read()
	contents = filedata.decode('utf-8')
	json_content=json.loads(contents)
	print(json_content)</pre>
	dove `json_content` contiene il dizionario Python.
		Sviluppatore di app, competenze in Python

Operazione	Description	Competenze richieste

Pulisci i valori dei dati per spazi e caratteri Unicode.	Come misura di sicurezza per garantire la correttezza dei dati, utilizzate il codice seguente per eseguire un'operazione di cancellazione dei valori inseriti. `json_content` Questo codice rimuove i caratteri di spazio all'inizio e alla fine di ogni stringa. Utilizza anche il `replace` metodo per rimuovere gli spazi rigidi (non interrotti) (i `\xa0` caratteri).
	
	`for item in json_content:`
	`  fqn_name = unicodedata.normalize("NFKD",item["FqdnName"].replace("u'", "'").replace('\xa0', '').strip())`
	`  rec_type = item["RecordType"].replace('\xa0', '').strip()`
	`  res_rec = {`
	
	`  'Value': item["Value"].replace('\xa0', '').strip()`
	`  }`
		Sviluppatore di app, competenze in Python

Operazione	Description	Competenze richieste
Inserisci record.	Utilizzate il codice seguente come parte del `for` ciclo precedente. `change_response = route53_client.change_resource_record_sets( HostedZoneId="xxxxxxxx", ChangeBatch={ 'Comment': 'Created by automation', 'Changes': [ { 'Action': 'UPSERT', 'ResourceRecordSet': { 'Name': fqn_name, 'Type': rec_type, 'TTL': item["TTL"], 'ResourceRecords': res_rec } } ] } )`	Sviluppatore di app, competenze in Python

Operazione	Description	Competenze richieste
	xxxxxxxDov'è l'ID della zona ospitata del primo passaggio di questa epopea.	

Risorse correlate

Riferimenti

- Creazione di record importando un file di zona (documentazione di Amazon Route 53)
- metodo create_hosted_zone (documentazione Boto3)
- metodo change_resource_record_sets (documentazione Boto3)

Tutorial e video

- Il tutorial su Python (documentazione Python)
- Progettazione DNS con Amazon Route 53 (YouTube video, AWS Online Tech Talks)

Modifica le intestazioni HTTP durante la migrazione da F5 a un Application Load Balancer su AWS

Sachin Trivedi, Amazon Web Services

Riepilogo

Quando esegui la migrazione di un'applicazione che utilizza un Load balancer F5 su Amazon Web Services (AWS) e desideri utilizzare un Application Load Balancer su AWS, la migrazione delle regole F5 per le modifiche agli header è un problema comune. Un Application Load Balancer non supporta le modifiche agli header, ma puoi usare Amazon CloudFront come rete per la distribuzione di contenuti (CDN) e Lambda @Edge per modificare le intestazioni.

Questo modello descrive le integrazioni richieste e fornisce codice di esempio per la modifica dell'intestazione utilizzando AWS e CloudFront Lambda @Edge.

Prerequisiti e limitazioni

Prerequisiti

- Un'applicazione locale che utilizza un sistema di bilanciamento del carico F5 con una configurazione che sostituisce il valore dell'intestazione HTTP utilizzando. `if, else` Per ulteriori informazioni su questa configurazione, vedete HTTP: :header nella documentazione del prodotto F5.

Limitazioni

- Questo modello si applica alla personalizzazione dell'intestazione del bilanciamento del carico F5. Per altri sistemi di bilanciamento del carico di terze parti, consulta la documentazione del sistema di bilanciamento del carico per informazioni di supporto.
- Le funzioni Lambda utilizzate per Lambda @Edge devono trovarsi nella regione Stati Uniti orientali (Virginia settentrionale).

Architecture

Il diagramma seguente mostra l'architettura su AWS, incluso il flusso di integrazione tra la CDN e altri componenti AWS.

Tools (Strumenti)

Servizi AWS

- Application Load Balancer – Un Application Load Balancer è un servizio di bilanciamento del carico AWS completamente gestito che funziona al settimo livello del modello Open Systems Interconnection (OSI). Bilancia il traffico su più destinazioni e supporta richieste di routing avanzate basate su intestazioni e metodi HTTP, stringhe di query e routing basato su host o percorsi.
- Amazon CloudFront: Amazon CloudFront è un servizio web che velocizza la distribuzione di contenuti web statici e dinamici, come .html, .css, .js e file di immagine, ai tuoi utenti. CloudFront distribuisce i tuoi contenuti attraverso una rete mondiale di data center denominati edge location per una latenza inferiore e prestazioni migliorate.
- Lambda @Edge – Lambda @Edge è un'estensione di AWS Lambda che consente di eseguire funzioni per personalizzare i contenuti distribuiti. CloudFront Puoi creare funzioni nella regione Stati Uniti orientali (Virginia settentrionale) e quindi associare la funzione a una CloudFront distribuzione per replicare automaticamente il codice in tutto il mondo, senza dover fornire o gestire server. Ciò riduce la latenza e migliora l'esperienza utente.

Codice

Il codice di esempio seguente fornisce un modello per modificare le intestazioni di risposta. CloudFront Segui le istruzioni nella sezione Epics per distribuire il codice.

```
exports.handler = async (event, context) => {
    const response = event.Records[0].cf.response;
    const headers = response.headers;


    const headerNameSrc = 'content-security-policy';
    const headerNameValue = '*.xyz.com';


    if (headers[headerNameSrc.toLowerCase()]) {
        headers[headerNameSrc.toLowerCase()] = [{
            key: headerNameSrc,
            value: headerNameValue,
        }];
        console.log(`Response header "${headerNameSrc}" was set to ` +
                    `"${headers[headerNameSrc.toLowerCase()][0].value}"`);
    }
```

```
    else {
            headers[headerNameSrc.toLowerCase()] = [{
            key: headerNameSrc,
            value: headerNameValue,
            }];
    }
    return response;
};
```

Epiche

Crea una distribuzione CDN

Operazione	Description	Competenze richieste
Crea una distribuzione CloudFront web.	In questo passaggio, crei una CloudFront distribuzione per indicare da CloudFront dove desideri che vengano distribuiti i contenuti e i dettagli su come monitorare e gestire la distribuzione dei contenuti. Per creare una distribuzione utilizzando la console, accedi alla Console di gestione AWS, apri la CloudFront console e segui i passaggi indicati nella CloudFront documentazione.	Amministratore del cloud

Crea e distribuisci la funzione Lambda @Edge

Operazione	Description	Competenze richieste
Crea e distribuisci una funzione Lambda @Edge.	È possibile creare una funzione Lambda @Edge utilizzando un blueprint per modificare CloudFront le	Amministratore AWS

Operazione	Description	Competenze richieste
	intestazioni di risposta. (Sono disponibili altri BluePrint per diversi casi d'uso; per ulteriori informazioni, consulta le funzioni di esempio di Lambda @Edge CloudFront nella documentazione.) Per creare una funzione Lambda @Edge: 1. Accedi alla Console di gestione AWS e apri la console AWS Lambda all'indirizzo. https://console.aws.amazon.com/lambda/ 2. Assicurati di trovarti nella regione Stati Uniti orientali (Virginia settentrionale). CloudFront i progetti sono disponibili solo in questa regione. 3. Scegli Crea funzione. 4. Scegli Usa un blueprint, quindi inserisci cloudfron t nel campo di ricerca Blueprints. 5. Scegli il cloudfront-modify-response-headerblueprint, quindi scegli Configura. 6. Nella pagina delle informazi oni di base, inserisci le seguenti informazioni:	

Operazione	Description	Competenze richieste
	a. Immettere il nome di una funzione. b. In Execution role (Ruolo di esecuzione) scegliere Create a new role from AWS policy templates (Crea nuovo ruolo dai modelli di policy AWS). c. Associa il nome del ruolo AWS Identity and Access Management (IAM) richiesto. 7. Scegli Crea funzione. 8. Nella sezione Designer della pagina, scegli il nome della funzione. 9. Nella sezione Codice funzionale, sostituisci il codice del modello con il codice di esempio fornito in precedenza in questo modello, nella sezione Codice. 10. Nel codice di esempio, sostituiscilo `xyz.com` con il tuo nome di dominio. 11. Scegli Save (Salva).	

Operazione	Description	Competenze richieste
Implementa la funzione Lambda @Edge.	Segui le istruzioni nel passaggio 4 del Tutorial: Creazione di una semplice funzione Lambda @Edge nella CloudFront documentazione di Amazon per configurare il CloudFront trigger e distribuire la funzione.	Amministratore AWS

Risorse correlate

CloudFront documentazione

- Comportamento di richiesta e risposta per origini personalizzate
- Lavorare con le distribuzioni
- Funzioni di esempio Lambda @Edge
- Personalizzazione all'edge con Lambda @Edge
- Tutorial: creazione di una semplice funzione Lambda @Edge

Crea un rapporto sui risultati di Network Access Analyzer per l'accesso a Internet in entrata in più Account AWS

Mike Virgilio, Amazon Web Services

Riepilogo

L'accesso involontario alle AWS risorse via Internet in entrata può comportare rischi per il perimetro dei dati di un'organizzazione. Network Access Analyzer è una funzionalità di Amazon Virtual Private Cloud (Amazon VPC) che ti aiuta a identificare accessi di rete non intenzionali alle tue risorse su Amazon Web Services ().AWS Puoi utilizzare Network Access Analyzer per specificare i requisiti di accesso alla rete e identificare potenziali percorsi di rete che non soddisfano i requisiti specificati. È possibile utilizzare Network Access Analyzer per effettuare le seguenti operazioni:

1. Identifica AWS le risorse accessibili a Internet tramite gateway Internet.
2. Verifica che i tuoi cloud privati virtuali (VPCs) siano segmentati in modo appropriato, ad esempio isolando gli ambienti di produzione e sviluppo e separando i carichi di lavoro transazionali.

Network Access Analyzer analizza le condizioni di raggiungibilità della rete e non solo un singolo componente end-to-end. Per determinare se una risorsa è accessibile a Internet, Network Access Analyzer valuta il gateway Internet, le tabelle di routing VPC, gli elenchi di controllo degli accessi alla rete (ACLs), gli indirizzi IP pubblici su interfacce di rete elastiche e i gruppi di sicurezza. Se qualcuno di questi componenti impedisce l'accesso a Internet, Network Access Analyzer non genera alcun risultato. Ad esempio, se un'istanza Amazon Elastic Compute Cloud (Amazon EC2) ha un gruppo di sicurezza aperto che consente il traffico proveniente da `0/0` una sottorete privata che non è instradabile da alcun gateway Internet, Network Access Analyzer non genererebbe alcun risultato. Ciò fornisce risultati ad alta fedeltà che consentono di identificare le risorse realmente accessibili da Internet.

Quando si esegue Network Access Analyzer, si utilizzano Network Access Scopes per specificare i requisiti di accesso alla rete. Questa soluzione identifica i percorsi di rete tra un gateway Internet e un'interfaccia di rete elastica. In questo modello, la soluzione viene distribuita in modo centralizzato Account AWS all'interno dell'organizzazione, gestita da AWS Organizations e analizza tutti gli account, in qualsiasi momento Regione AWS, dell'organizzazione.

Questa soluzione è stata progettata tenendo presente quanto segue:

- I AWS CloudFormation modelli riducono lo sforzo richiesto per distribuire le AWS risorse secondo questo schema.
- È possibile modificare i parametri nei CloudFormation modelli e nello script naa-script.sh al momento della distribuzione per personalizzarli in base all'ambiente.
- Lo scripting di Bash fornisce e analizza automaticamente gli ambiti di accesso alla rete per più account, in parallelo.
- Uno script Python elabora i risultati, estrae i dati e quindi consolida i risultati. È possibile scegliere di esaminare il rapporto consolidato dei risultati di Network Access Analyzer in formato CSV o in formato. AWS Security Hub CSPM Un esempio del rapporto CSV è disponibile nella sezione Informazioni aggiuntive di questo modello.
- È possibile correggere i risultati oppure escluderli dalle analisi future aggiungendoli al file naa-exclusions.csv.

Prerequisiti e limitazioni

Prerequisiti

- E Account AWS per ospitare servizi e strumenti di sicurezza, gestiti come account membro di un'organizzazione in. AWS Organizations In questo schema, questo account viene definito account di sicurezza.
- Nell'account di sicurezza, è necessario disporre di una sottorete privata con accesso a Internet in uscita. Per istruzioni, consulta Creare una sottorete nella documentazione di Amazon VPC. È possibile stabilire l'accesso a Internet utilizzando un gateway NAT o un endpoint VPC di interfaccia.
- Accesso all'account di AWS Organizations gestione o a un account con autorizzazioni di amministratore delegate per. CloudFormation Per istruzioni, consulta Registrare un amministratore delegato nella documentazione. CloudFormation
- Abilita l'accesso affidabile tra AWS Organizations e CloudFormation. Per istruzioni, consulta Abilita l'accesso affidabile con AWS Organizations nella CloudFormation documentazione.
- Se stai caricando i risultati su Security Hub CSPM, Security Hub CSPM deve essere abilitato nell'account e nel luogo in cui viene fornita l' Regione AWS istanza Amazon. EC2 Per ulteriori informazioni, consulta Configurazione AWS Security Hub CSPM.

Limitazioni

- I percorsi di rete tra account non vengono attualmente analizzati a causa delle limitazioni della funzionalità Network Access Analyzer.
- L'obiettivo Account AWS deve essere gestito come organizzazione in. AWS OrganizationsSe non lo utilizzi AWS Organizations, puoi aggiornare il CloudFormation modello naa-execrole.yaml e lo script naa-script.sh per il tuo ambiente. Fornisci invece un elenco delle regioni Account AWS IDs e delle regioni in cui desideri eseguire lo script.
- Il CloudFormation modello è progettato per distribuire l' EC2 istanza Amazon in una sottorete privata con accesso a Internet in uscita. L' AWS Systems Manager agente (agente SSM) richiede l'accesso in uscita per raggiungere l'endpoint del servizio Systems Manager ed è necessario l'accesso in uscita per clonare l'archivio del codice e installare le dipendenze. Se desideri utilizzare una sottorete pubblica, devi modificare il modello naa-resources.yaml per associare un indirizzo IP elastico all'istanza Amazon. EC2

Architecture

Architettura di Target

Opzione 1: accedere ai risultati in un bucket Amazon S3

Il diagramma mostra il seguente processo:

1. Se esegui manualmente la soluzione, l'utente si autentica sull' EC2 istanza Amazon utilizzando Session Manager e quindi esegue lo script naa-script.sh. Questo script di shell esegue i passaggi da 2 a 7.

   Se esegui automaticamente la soluzione, lo script naa-script.sh si avvia automaticamente in base alla pianificazione definita nell'espressione cron. Questo script di shell esegue i passaggi da 2 a 7. Per ulteriori informazioni, vedere Automazione e scalabilità alla fine di questa sezione.
2. L' EC2 istanza Amazon scarica il file naa-exception.csv più recente dal bucket Amazon S3. Questo file viene utilizzato più avanti nel processo quando lo script Python elabora le esclusioni.
3. L' EC2 istanza Amazon assume il ruolo `NAAEC2Role` AWS Identity and Access Management (IAM), che concede le autorizzazioni per accedere al bucket Amazon S3 e per assumere i ruoli `NAAExecRole` IAM negli altri account dell'organizzazione.
4. L' EC2 istanza Amazon assume il ruolo `NAAExecRole` IAM nell'account di gestione dell'organizzazione e genera un elenco degli account dell'organizzazione.

5. L' EC2 istanza Amazon assume il ruolo `NAAExecRole` IAM negli account dei membri dell'organizzazione (chiamati account di carico di lavoro nel diagramma dell'architettura) ed esegue una valutazione della sicurezza in ogni account. I risultati vengono archiviati come file JSON sull' EC2 istanza Amazon.
6. L' EC2 istanza Amazon utilizza uno script Python per elaborare i file JSON, estrarre i campi di dati e creare un report CSV.
7. L' EC2 istanza Amazon carica il file CSV nel bucket Amazon S3.
8. Una EventBridge regola Amazon rileva il caricamento del file e utilizza un argomento Amazon SNS per inviare un'e-mail che notifica all'utente che il rapporto è completo.
9. L'utente scarica il file CSV dal bucket Amazon S3. L'utente importa i risultati nel modello Excel e li esamina.

Opzione 2: Accedi ai risultati in AWS Security Hub CSPM

Il diagramma mostra il seguente processo:

1. Se esegui manualmente la soluzione, l'utente si autentica sull' EC2 istanza Amazon utilizzando Session Manager e quindi esegue lo script naa-script.sh. Questo script di shell esegue i passaggi da 2 a 7.

   Se esegui automaticamente la soluzione, lo script naa-script.sh si avvia automaticamente in base alla pianificazione definita nell'espressione cron. Questo script di shell esegue i passaggi da 2 a 7. Per ulteriori informazioni, vedere Automazione e scalabilità alla fine di questa sezione.
2. L' EC2 istanza Amazon scarica il file naa-exception.csv più recente dal bucket Amazon S3. Questo file viene utilizzato più avanti nel processo quando lo script Python elabora le esclusioni.
3. L' EC2 istanza Amazon assume il ruolo `NAAEC2Role` IAM, che concede le autorizzazioni per accedere al bucket Amazon S3 e per assumere i ruoli `NAAExecRole` IAM negli altri account dell'organizzazione.
4. L' EC2 istanza Amazon assume il ruolo `NAAExecRole` IAM nell'account di gestione dell'organizzazione e genera un elenco degli account dell'organizzazione.
5. L' EC2 istanza Amazon assume il ruolo `NAAExecRole` IAM negli account dei membri dell'organizzazione (chiamati account di carico di lavoro nel diagramma dell'architettura) ed esegue una valutazione della sicurezza in ogni account. I risultati vengono archiviati come file JSON sull' EC2 istanza Amazon.

6. L' EC2 istanza Amazon utilizza uno script Python per elaborare i file JSON ed estrarre i campi di dati per l'importazione in Security Hub CSPM.
7. L' EC2 istanza Amazon importa i risultati di Network Access Analyzer in Security Hub CSPM.
8. Una EventBridge regola Amazon rileva l'importazione e utilizza un argomento Amazon SNS per inviare un'e-mail che notifica all'utente che il processo è completo.
9. L'utente visualizza i risultati in Security Hub CSPM.

Automazione e scalabilità

È possibile pianificare questa soluzione per eseguire automaticamente lo script naa-script.sh secondo una pianificazione personalizzata. Per impostare una pianificazione personalizzata, nel modello CloudFormation naa-resources.yaml, modifica il parametro. `CronScheduleExpression` Ad esempio, il valore predefinito di `0 0 * * 0` esegue la soluzione a mezzanotte di ogni domenica. Il valore di `0 0 * 1-12 0` eseguirebbe la soluzione a mezzanotte della prima domenica di ogni mese. Per ulteriori informazioni sull'uso delle espressioni cron, vedere Cron e rate expression nella documentazione di Systems Manager.

Se desideri modificare la pianificazione dopo che lo `NAA-Resources` stack è stato distribuito, puoi modificare manualmente la pianificazione cron in. `/etc/cron.d/naa-schedule`

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altro modo. Account AWS
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Organizationsè un servizio di gestione degli account che ti aiuta a consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente.
- AWS Security Hub CSPMfornisce una visione completa dello stato di sicurezza in AWS. Inoltre, consente di verificare la conformità AWS dell'ambiente agli standard e alle best practice del settore della sicurezza.

- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Systems Managerti aiuta a gestire le applicazioni e l'infrastruttura in esecuzione in. Cloud AWS Semplifica la gestione delle applicazioni e delle risorse, riduce i tempi di rilevamento e risoluzione dei problemi operativi e aiuta a gestire le AWS risorse in modo sicuro su larga scala. Questo modello utilizza Session Manager, una funzionalità di Systems Manager.

Archivio di codice

Il codice di questo pattern è disponibile nel repository GitHub Network Access Analyzer Multi-Account Analysis. L'archivio del codice contiene i seguenti file:

- naa-script.sh — Questo script bash viene utilizzato per avviare un'analisi di più Network Access Analyzer Account AWS, in parallelo. Come definito nel CloudFormation modello naa-resources.yaml, questo script viene distribuito automaticamente nella cartella sull'istanza Amazon. `/usr/local/naa` EC2
- naa-resources.yaml: utilizzi questo modello per creare uno stack nell'account di sicurezza dell'organizzazione. CloudFormation Questo modello distribuisce tutte le risorse necessarie per questo account per supportare la soluzione. Questo stack deve essere distribuito prima del modello naa-execrole.yaml.

> **Note**
>
> Se questo stack viene eliminato e ridistribuito, è necessario ricostruire il set di stack per ricostruire le dipendenze tra account tra i `NAAExecRole` ruoli IAM.

- naa-execrole.yaml: utilizzi questo CloudFormation modello per creare un set di stack che distribuisce il ruolo IAM in tutti gli account dell'organizzazione, incluso l'account di gestione. `NAAExecRole`
- naa-processfindings.py — Lo script naa-script.sh chiama automaticamente questo script Python per elaborare gli output JSON di Network Access Analyzer, escludere eventuali risorse note valide nel file naa-exclusions.csv e quindi generare un file CSV dei risultati consolidati o importare i risultati in Security Hub CSPM.

Epiche

Preparati per l'implementazione

Operazione	Description	Competenze richieste
Clona il repository del codice.	1. In un'interfaccia a riga di comando, modificate la directory di lavoro nella posizione in cui desiderate archiviare i file di esempio. 2. Immetti il comando seguente: `git clone https://github.com/aws-samples/network-access-analyzer-multi-account-analysis.git`	AWS DevOps
Esamina i modelli.	1. Nel repository clonato, apri i file naa-resources.yaml e naa-execrole.yaml. 2. Esamina le risorse create da questi modelli e modifica i modelli in base alle esigenze del tuo ambiente. Per ulteriori informazioni, consulta Lavorare con i modelli nella CloudFormation documentazione. 3. Salvate e chiudete i file naa-resources.yaml e naa-execrole.yaml.	AWS DevOps

Crea gli CloudFormation stack

Operazione	Description	Competenze richieste
Fornisci risorse nell'account di sicurezza.	Utilizzando il modello naa-resources.yaml, crei uno CloudFormation stack che distribuisce tutte le risorse richieste nell'account di sicurezza. Per istruzioni, consulta Creazione di uno stack nella documentazione. CloudFormation Tieni presente quanto segue durante la distribuzione di questo modello: 1. Nella pagina Specificare il modello, scegli Il modello è pronto, quindi carica il file naa-resources.yaml. 2. Nella pagina Specificare i dettagli dello stack, nella casella Nome dello stack, immettere. `NAA-Resources` 3. Nella sezione Parametri, inserisci quanto segue: • `VPCId`— Seleziona un VPC nell'account. • `SubnetId`— Seleziona una sottorete privata con accesso a Internet.	AWS DevOps

Operazione	Description	Competenze richieste
	Note Se selezioni una sottorete pubblica, all' EC2 istanza Amazon potrebbe non essere assegnato un indirizzo IP pubblico perché il CloudForm ation modello, per impostazi one predefini ta, non fornisce e non collega un indirizzo IP elastico. • `InstanceType` — Lascia il tipo di istanza predefinito. • `InstanceImageId` — Lasciare il valore predefinito. • `KeyPairName` — Se utilizzi SSH per l'accesso , specifica il nome di una coppia di key pair esistente. • `PermittedSSHInboun d` — Se utilizzi SSH per l'accesso, specifica un blocco CIDR	

Operazione	Description	Competenze richieste
	consentito. Se non utilizzi SSH, mantieni il valore predefinito di. `127.0.0.1` • `BucketName` — Il valore predefinito è`naa-<accountID>-<region>` . È possibile modificarlo in base alle esigenze. Se si specifica un valore personalizzato, l'ID dell'account e la regione vengono aggiunti automaticamente al valore specificato. • `EmailAddress` — Specificare un indirizzo e-mail per una notifica Amazon SNS al termine dell'analisi. **Note** La configurazione dell'abbonamento Amazon SNS deve essere confermata prima del completamento dell'analisi, altrimenti non verrà inviata alcuna notifica.	

Operazione	Description	Competenze richieste
	• `NAAEC2Role` — Mantieni l'impostazione predefinita a meno che le convenzioni di denominazione non richiedano un nome diverso per questo ruolo IAM. • `NAAExecRole` — Mantieni il valore predefinito a meno che non venga utilizzato un altro nome durante la distribuzione di naa-execrole.yaml • `Parallelism` — Specificare il numero di valutazioni parallele da eseguire. • `Regions`— Specificare Regioni AWS quello che si desidera analizzare. • `ScopeNameValue` — Specificare il tag che verrà assegnato all'ambito. Questo tag viene utilizzato per determinare l'ambito di accesso alla rete. • `ExclusionFile` — Specificare il nome del file di esclusione. Le voci in questo file verranno escluse dai risultati.	

Operazione	Description	Competenze richieste
	• `FindingsToCSV` — Specificare se i risultati devono essere esportati in formato CSV. I valori accettati sono `true` e. `false` • `FindingsToSecurityHub` — Specificare se i risultati devono essere importati in Security Hub CSPM. I valori accettati sono `true` e. `false` • `EmailNotificationsForSecurityHub` — Specificare se l'importazione dei risultati in Security Hub CSPM deve generare notifiche e-mail. I valori accettati sono e. `true` `false` • `ScheduledAnalysis` — Se desiderate che la soluzione venga eseguita automaticamente in base a una pianificazione`true`, immettete e quindi personalizzate la pianificazione nel `CronScheduleExpression` parametro. Se non desiderate eseguire la soluzione automatic	

Operazione	Description	Competenze richieste
	amente, immettete `false`. • `CronScheduleExpression` — Se state eseguendo la soluzione automaticamente, inserite un'espressione cron per definire la pianificazione. Per ulteriori informazioni, consulta Automazione e scalabilità nella sezione Architettura di questo modello. 1. Nella pagina Revisione , seleziona Le seguenti risorse richiedono funzionalità: [AWS::IAM::Role], quindi scegli Create Stack. 2. Dopo che lo stack è stato creato correttamente, nella CloudFormation console, nella scheda Outputs, copia l'`NAAEC2Role` Amazon Resource Name (ARN). Utilizzerai questo ARN più tardi quando distribuirai il file naa-execrole.yaml.	

Operazione	Description	Competenze richieste
Fornisci il ruolo IAM negli account dei membri.	Nell'account di AWS Organizations gestione o in un account con autorizzazioni di amministratore delegato per CloudFormation, utilizza il modello naa-execrole.yaml per creare un set di stack. CloudFormation Il set di stack implementa il ruolo IAM in tutti gli account membri dell'organizzazione. `NAAExecRole` Per istruzioni, consulta Creare un set di stack con autorizzazioni gestite dal servizio nella documentazione. CloudFormation Tieni presente quanto segue durante la distribuzione di questo modello: 1. In Prepara modello, scegli Il modello è pronto, quindi carica il file naa-execrole.yaml. 2. Nella pagina Specificare i dettagli, assegnate un nome al set di stack StackSet . `NAA-ExecRole` 3. Nella sezione Parametri, inserisci quanto segue: • `AuthorizedARN` — Inserisci l'`NAAEC2Role` ARN, che hai copiato	AWS DevOps

Operazione	Description	Competenze richieste
	quando hai creato lo stack. `NAA-Resources` • `NAARoleName` — Mantieni il valore predefinito a **`NAAExecRole`** meno che non sia stato usato un altro nome durante la distribuzione del file naa-resources.yaml. 4. In Permissions (Autorizzazioni) scegliere Service-managed permissions (Autorizzazioni gestite dal servizio). 5. Nella pagina Imposta opzioni di distribuzione, in Obiettivi di distribuzione, scegli Distribuisci nell'organizzazione e accetta tutte le impostazioni predefinite. **Note** Se desideri che gli stack vengano distribuiti contemporaneamente a tutti gli account membri, imposta il numero massimo di account simultanei e la tolleranza agli errori su un	

Operazione	Description	Competenze richieste
	valore elevato, ad esempio. `100` 6. In Regioni di distribuzione, scegli la regione in cui viene distribuita l' EC2 istanza Amazon per Network Access Analyzer. Poiché le risorse IAM sono globali e non regionali, questo implementa il ruolo IAM in tutte le regioni attive. 7. Nella pagina di revisione , seleziona Riconosco che AWS CloudFormation potrebbe creare risorse IAM con nomi personalizzati, quindi scegli Crea StackSet. 8. Monitora la scheda Stack Instances (per lo stato dei singoli account) e la scheda Operations (per lo stato generale) per determinare quando la distribuzione è completa.	

Operazione	Description	Competenze richieste
Fornisci il ruolo IAM nell'account di gestione.	Utilizzando il modello naa-execrole.yaml, crei uno CloudFormation stack che distribuisce il ruolo IAM nell'account di gestione dell'`NAAExecRole` organizzazione. Lo stack set creato in precedenza non distribuisce il ruolo IAM nell'account di gestione. Per istruzioni, consulta Creazione di uno stack nella documentazione. CloudFormation Tieni presente quanto segue durante la distribuzione di questo modello: 1. Nella pagina Specificare il modello, scegliete Il modello è pronto, quindi caricate il file naa-execrole.yaml. 2. Nella pagina Specificare i dettagli dello stack, nella casella Nome dello stack, immettere. `NAA-ExecRole` 3. Nella sezione Parametri, inserisci quanto segue: • `AuthorizedARN` — Inserisci l'`NAAEC2Role` ARN, che hai copiato quando hai creato lo stack. `NAA-Resources`	AWS DevOps

Operazione	Description	Competenze richieste
	• `NAARoleName` — Mantieni il valore predefinito a **`NAAExecRole`** meno che non sia stato usato un altro nome durante la distribuzione del file naa-resources.yaml. 4. Nella pagina Revisione, seleziona Le seguenti risorse richiedono funzionalità: [], quindi scegli Crea stack. AWS::IAM::Role	

Esegui l'analisi

Operazione	Description	Competenze richieste
Personalizza lo script della shell.	1. Accedi all'account di sicurezza dell'organizzazione. 2. Utilizzando Session Manager, connettiti all'EC2 istanza Amazon per Network Access Analyzer di cui hai precedentemente fornito il provisioning. Per istruzioni, consulta Connect alla tua istanza Linux usando Session Manager. Se non riesci a connetterti, consulta la sezione Risoluzione dei problemi di questo schema.	AWS DevOps

Operazione	Description	Competenze richieste

	3. Immettete i seguenti comandi per aprire il file naa-script.sh e modificarlo:
	
	`sudo -i`
	`cd /usr/local/naa`
	`vi naa-script.sh`
	
	4. Esaminate e modificate i parametri e le variabili regolabili in questo script in base alle esigenze del vostro ambiente. Per ulteriori informazioni sulle opzioni di personali zzazione, consultate i commenti all'inizio dello script.
	
	Ad esempio, anziché ottenere un elenco di tutti gli account dei membri dell'organizzazione dall'account di gestione, è possibile modificare lo script per specificare l' Account AWS IDs o Regioni AWS che si desidera scansiona re oppure fare riferimento a un file esterno che contiene questi parametri.
	
	5. Salvate e chiudete il file naa-script.sh.	

Operazione	Description	Competenze richieste

Analizza gli account target.	1. Esegui i comandi seguenti: Questo esegue lo script naa-script.sh:
	`sudo -i`
	`cd /usr/local/naa`
	`screen`
	`./naa-script.sh`
	Tenere presente quanto segue:
	• Il `screen` comando consente allo script di continuare l'esecuzione nel caso in cui la connessione scada o si perda l'accesso alla console.
	• Dopo l'avvio della scansione, puoi forzare il distacco dello schermo premendo Ctrl+A D. La schermata si stacca ed è possibile chiudere la connessione dell'istanza mentre l'analisi procede.
	• Per riprendere una sessione separata, connettiti all'istanza, entra e poi accedi. `sudo -i screen -r`
	2. Monitorate l'output per eventuali errori per assicurarvi che lo script
		AWS DevOps

Operazione	Description	Competenze richieste
	funzioni correttamente. Per un esempio di output, consultate la sezione Informazioni aggiuntive di questo modello. 3. Attendi il completamento del processo. Se hai configurato le notifiche e-mail, riceverai un'e-mail quando i risultati sono stati caricati nel bucket Amazon S3 o importati in Security Hub CSPM.	
Opzione 1: recupera i risultati dal bucket Amazon S3.	1. Scarica il file CSV dal bucket. `naa-<accountID>-<region>` Per istruzioni, consulta Download di un oggetto nella documentazione di Amazon S3. 2. Elimina il file CSV dal bucket Amazon S3. Si tratta di una best practice per l'ottimizzazione dei costi. Per istruzioni, consulta Eliminazione di oggetti nella documentazione di Amazon S3.	AWS DevOps

Operazione	Description	Competenze richieste
Opzione 2: rivedi i risultati in Security Hub CSPM.	1. Apri la console Security Hub CSPM. 2. Scegli Findings dal pannello di navigazione. 3. Esamina i risultati di Network Access Analyzer. Per istruzioni, consulta Visualizzazione degli elenchi e dei dettagli dei risultati nella documenta zione CSPM di Security Hub. **Note** È possibile cercare i risultati aggiungen do un titolo che inizia con il filtro e l'immissione. `Network Access Analyzer`	AWS DevOps

Correggi ed escludi i risultati

Operazione	Description	Competenze richieste
Correggere i risultati.	Correggi i risultati che desideri correggere. Per ulteriori informazioni e best practice su come creare un perimetro attorno a AWS identità, risorse e reti, consulta Building a data	AWS DevOps

Operazione	Description	Competenze richieste
	perimeter on (Whitepaper). AWSAWS	

Operazione	Description	Competenze richieste
Escludi risorse con percorsi di rete noti e validi.	Se Network Access Analyzer genera risultati relativi a risorse che dovrebbero essere accessibili da Internet, puoi aggiungere tali risorse a un elenco di esclusione. La prossima volta che Network Access Analyzer verrà eseguito, non genererà alcun risultato per quella risorsa. 1. Accedere allo script naa-script.sh **`/usr/local/naa`** , quindi aprirlo. Prendi nota del valore della `S3_EXCLUSION_FILE` variabile. 2. Se il valore della `S3_EXCLUSION_FILE` variabile è`true`, scarica il file naa-exclusions.csv dal `naa-<accountID>-<region>` bucket. Per istruzioni, consulta Download di un oggetto nella documentazione di Amazon S3. Se il valore della `S3_EXCLUSION_FILE` variabile è`false`, accedi al file naa-exclusions.csv **`/usr/local/naa`** e apri.	AWS DevOps

Operazione	Description	Competenze richieste

	Note
	Se il valore della `S3_EXCLUSION_FILE` variabile è`false`, lo script utilizza una versione locale del file delle esclusioni. Se successivamente modifichi il valore in`true`, lo script sovrascrive la versione locale con il file nel bucket Amazon S3.
	3. Nel file naa-exclusions.csv, inserisci le risorse che desideri escludere. Immettete una risorsa in ogni riga e utilizzate il formato seguente.
	`<resource_id>,<secgroup_id>,<sgrule_cidr>,<sgrule_portrange>,<sgrule_protocol>`
	Di seguito è riportato un esempio di risorsa.
	`eni-1111aaaaa2222bbbb,sg-3333ccccc44`	

Operazione	Description	Competenze richieste
	`44dddd,0.0.0.0/0,80 to 80,tcp` 4. Salvate e chiudete il file naa-exclusions.csv. 5. Se hai scaricato il file naa-exclusions.csv dal bucket Amazon S3, carica la nuova versione. Per istruzioni, consulta Caricamento di oggetti nella documentazione di Amazon S3.	

(Facoltativo) Aggiorna lo script naa-script.sh

Operazione	Description	Competenze richieste
Aggiornare lo script naa-script.sh.	Se desideri aggiornare lo script naa-script.sh all'ultima versione del repository, procedi come segue: 1. Connect all' EC2 istanza Amazon utilizzando Session Manager. Per istruzioni, consulta Connect alla tua istanza Linux usando Session Manager. 2. Immetti il comando seguente: `sudo -i` 3. Passa alla directory degli script naa-script.sh:	AWS DevOps

Operazione	Description	Competenze richieste

	`cd /usr/local/naa`
	4. Immettete il seguente comando per nasconder e lo script locale in modo da poter unire le modifiche personalizzate nella versione più recente:
	`git stash`
	5. Immettete il seguente comando per ottenere la versione più recente dello script:
	`git pull`
	6. Immettete il seguente comando per unire lo script personalizzato alla versione più recente dello script:
	`git stash pop`	

(Facoltativo) Pulizia

Operazione	Description	Competenze richieste
Eliminare tutte le risorse distribuite.	È possibile lasciare le risorse distribuite negli account. Se desideri eseguire il deprovisioning di tutte le risorse, procedi come segue:	AWS DevOps

Operazione	Description	Competenze richieste
	1. Elimina lo `NAA-ExecRole` stack fornito nell'account di gestione. Per istruzioni, consulta Eliminazione di uno stack nella documentazione. CloudFormation 2. Elimina il set di `NAA-ExecRole` stack fornito nell'account di gestione dell'organizzazione o nell'account amministratore delegato. Per istruzioni, consulta Eliminare un set di stack nella documentazione. CloudFormation 3. Elimina tutti gli oggetti nel `naa-<accountID>-<region>` bucket Amazon S3. Per istruzioni, consulta Eliminazione di oggetti nella documentazione di Amazon S3. 4. Elimina lo `NAA-Resources` stack fornito nell'account di sicurezza. Per istruzioni, consulta Eliminazione di uno stack nella documentazione. CloudFormation	

risoluzione dei problemi

Problema	Soluzione
Impossibile connettersi all' EC2 istanza Amazon utilizzando Session Manager.	L'agente SSM deve essere in grado di comunicare con l'endpoint Systems Manager. Esegui questa operazione: 1. Verifica che la sottorete in cui è distribuita l' EC2 istanza Amazon disponga dell'accesso a Internet. 2. Riavvia l' EC2 istanza Amazon.
Quando distribuisci lo stack set, la CloudForm ation console ti chiede di farlo. `Enable trusted access with AWS Organizat ions to use service-managed permissions`	Ciò indica che l'accesso affidabile non è stato abilitato tra e. AWS Organizations CloudForm ation È necessario un accesso affidabile per distribuire il set di stack gestito dai servizi. Scegli il pulsante per abilitare l'accesso affidabil e. Per ulteriori informazioni, consulta Abilitare l'accesso affidabile nella CloudFormation documentazione.

Risorse correlate

- Novità: Amazon VPC Network Access Analyzer (AWS post di blog)
- AWS re:Inforce 2022 - Convalida controlli efficaci di accesso alla rete su AWS (02) (video) NIS2
- Demo - Analisi del percorso dei dati di accesso a Internet a livello di organizzazione mediante Network Access Analyzer (video)

Informazioni aggiuntive

Esempio di output da console

L'esempio seguente mostra il risultato della generazione dell'elenco degli account di destinazione e dell'analisi degli account di destinazione.

```
[root@ip-10-10-43-82 naa]# ./naa-script.sh
```

```
download: s3://naa-<account ID>-us-east-1/naa-exclusions.csv to ./naa-exclusions.csv

AWS Management Account: <Management account ID>

AWS Accounts being processed...
<Account ID 1> <Account ID 2> <Account ID 3>

Assessing AWS Account: <Account ID 1>, using Role: NAAExecRole
Assessing AWS Account: <Account ID 2>, using Role: NAAExecRole
Assessing AWS Account: <Account ID 3>, using Role: NAAExecRole
Processing account: <Account ID 1> / Region: us-east-1
Account: <Account ID 1> / Region: us-east-1 - Detecting Network Analyzer scope...
Processing account: <Account ID 2> / Region: us-east-1
Account: <Account ID 2> / Region: us-east-1 - Detecting Network Analyzer scope...
Processing account: <Account ID 3> / Region: us-east-1
Account: <Account ID 3> / Region: us-east-1 - Detecting Network Analyzer scope...
Account: <Account ID 1> / Region: us-east-1 - Network Access Analyzer scope detected.
Account: <Account ID 1> / Region: us-east-1 - Continuing analyses with Scope ID.
 Accounts with many resources may take up to one hour
Account: <Account ID 2> / Region: us-east-1 - Network Access Analyzer scope detected.
Account: <Account ID 2> / Region: us-east-1 - Continuing analyses with Scope ID.
 Accounts with many resources may take up to one hour
Account: <Account ID 3> / Region: us-east-1 - Network Access Analyzer scope detected.
Account: <Account ID 3> / Region: us-east-1 - Continuing analyses with Scope ID.
 Accounts with many resources may take up to one hour
```

Esempi di report CSV

Le immagini seguenti sono esempi dell'output CSV.

Configura la risoluzione DNS per reti ibride in un ambiente multi-account AWS

Anvesh Koganti, Amazon Web Services

Riepilogo

Questo modello fornisce una soluzione completa per configurare la risoluzione DNS in ambienti di rete ibridi che includono più account Amazon Web Services (AWS). Consente la risoluzione DNS bidirezionale tra le reti locali e l'ambiente tramite gli endpoint. AWS Amazon Route 53 Resolver Il modello presenta due soluzioni per abilitare la risoluzione DNS in un'architettura centralizzata e multi-account:

- La configurazione di base non utilizza i profili Route 53. Aiuta a ottimizzare i costi per implementazioni di piccole e medie dimensioni con complessità inferiore.
- La configurazione avanzata utilizza i profili Route 53 per semplificare le operazioni. È ideale per implementazioni DNS più grandi o più complesse.

> **Note**
>
> Consulta la sezione Limitazioni per le limitazioni e le quote del servizio prima dell'implementazione. Prendi in considerazione fattori come le spese generali di gestione, i costi, la complessità operativa e l'esperienza del team quando prendi una decisione.

Prerequisiti e limitazioni

Prerequisiti

- Un ambiente AWS multi-account con Amazon Virtual Private Cloud (Amazon VPC) distribuito su Shared Services e account di carico di lavoro (preferibilmente configurato tramite AWS Control Tower AWS seguendo le best practice per la struttura degli account).
- Connettività ibrida esistente (AWS Direct Connect o AWS Site-to-Site VPN) tra la rete locale e l'ambiente. AWS
- Peering Amazon VPC o Cloud AWS WAN per la AWS Transit Gateway connettività di rete Layer 3 tra. VPCs (Questa connettività è necessaria per il traffico delle applicazioni. Non è necessario

per il funzionamento della risoluzione DNS. La risoluzione DNS funziona indipendentemente dalla connettività di rete tra.) VPCs

- Server DNS in esecuzione nell'ambiente locale.

Limitazioni

- Gli endpoint, le regole e i profili di Route 53 Resolver sono costrutti regionali e potrebbero richiedere la replica in più parti per le organizzazioni globali. Regioni AWS
- Per un elenco completo delle quote di servizio per Route 53 Resolver, zone ospitate private e profili, consulta Quotas nella documentazione di Route 53.

Architecture

Stack tecnologico Target

- Endpoint Route 53 in uscita e in entrata
- Regole di Route 53 Resolver per l'inoltro condizionale
- AWS Resource Access Manager (AWS RAM)
- Zona ospitata privata Route 53

Architettura Target

Endpoint in uscita e in entrata

Il diagramma seguente mostra il flusso di risoluzione DNS da a on-premise. AWS Questa è la configurazione della connettività per le risoluzioni in uscita in cui il dominio è ospitato in locale. Di seguito è riportata una panoramica di alto livello del processo necessario per configurarlo. Per i dettagli, consulta la sezione Epics.

1. Distribuisci gli endpoint Route 53 Resolver in uscita nel VPC di Shared Services.
2. Crea regole Route 53 Resolver (regole di inoltro) nell'account Shared Services per i domini ospitati in locale.
3. Condividi e associa le regole ad altri account che ospitano le risorse necessarie per risolvere i domini ospitati in locale. VPCs Questa operazione può essere eseguita in diversi modi a seconda del caso d'uso, come descritto più avanti in questa sezione.

Dopo aver configurato la connettività, i passaggi necessari per la risoluzione in uscita sono i seguenti:

1. L'istanza Amazon Elastic Compute Cloud (Amazon EC2) invia una richiesta di risoluzione DNS `db.onprem.example.com` al Route 53 Resolver del VPC all'indirizzo VPC+2.
2. Route 53 Resolver controlla le regole del Resolver e inoltra la richiesta al server DNS locale utilizzando l'endpoint in uscita. IPs
3. L'endpoint in uscita inoltra la richiesta al DNS locale. IPs Il traffico passa attraverso la connettività di rete ibrida stabilita tra il VPC di Shared Services e il data center locale.
4. Il server DNS locale risponde all'endpoint in uscita, che quindi inoltra la risposta al Route 53 Resolver del VPC. Il Resolver restituisce la risposta all'istanza. EC2

Il diagramma seguente mostra il flusso di risoluzione DNS dall'ambiente locale a. AWS Questa è la configurazione della connettività per le risoluzioni in entrata su cui è ospitato il dominio. AWS Di seguito è riportata una panoramica di alto livello del processo necessario per configurarlo. Per i dettagli, consulta la sezione Epics.

1. Distribuisci gli endpoint Resolver in ingresso nel VPC di Shared Services.
2. Crea zone private ospitate nell'account Shared Services (approccio centralizzato).
3. Associa le zone ospitate private al VPC di Shared Services. Condividi e associa queste zone a più account VPCs per la risoluzione VPC-to-VPC DNS. Questa operazione può essere eseguita in diversi modi a seconda del caso d'uso, come descritto più avanti in questa sezione.

Dopo aver configurato la connettività, i passaggi necessari per la risoluzione in entrata sono i seguenti:

1. La risorsa locale invia una richiesta di risoluzione DNS `ec2.prod.aws.example.com` al server DNS locale.
2. Il server DNS locale inoltra la richiesta all'endpoint Resolver in ingresso nel VPC di Shared Services tramite la connessione di rete ibrida.
3. L'endpoint Resolver in entrata cerca la richiesta nella zona ospitata privata associata con l'aiuto del Resolver VPC Route 53 e ottiene l'indirizzo IP appropriato.

4. Questi indirizzi IP vengono rispediti al server DNS locale, che restituisce la risposta alla risorsa locale.

Questa configurazione consente alle risorse locali di risolvere i nomi di dominio AWS privati instradando le query attraverso gli endpoint in entrata verso la zona ospitata privata appropriata. In questa architettura, le zone private ospitate sono centralizzate in un VPC di Shared Services, che consente la gestione centralizzata del DNS da parte di un singolo team. Queste zone possono essere associate a molte VPCs per risolvere il caso d'uso della risoluzione VPC-to-VPC DNS. In alternativa, potresti voler delegare la proprietà e la gestione del dominio DNS a ciascuna di esse. Account AWS In tal caso, ogni account gestisce le proprie zone ospitate private e associa ogni zona al VPC centrale di Shared Services per una risoluzione unificata con l'ambiente locale. Questo approccio decentralizzato non rientra nell'ambito di questo modello. Per ulteriori informazioni, consulta Scalare la gestione DNS su più account e VPCs nel white paper Hybrid Cloud DNS Options for Amazon VPC.

Quando stabilisci i flussi di risoluzione DNS fondamentali utilizzando gli endpoint Resolver, devi determinare come gestire la condivisione e l'associazione delle regole Resolver e delle zone ospitate private tra le tue. Account AWSÈ possibile affrontare questo problema in due modi: tramite la condivisione autogestita utilizzando le regole del Resolver AWS RAM per condividere e le associazioni dirette di zone ospitate private, come dettagliato nella sezione Configurazione di base, o tramite i profili Route 53, come discusso nella sezione Configurazione avanzata. La scelta dipende dalle preferenze di gestione DNS e dai requisiti operativi dell'organizzazione. I seguenti diagrammi di architettura illustrano un ambiente scalabile che include più account VPCs su diversi account, che rappresenta una tipica implementazione aziendale.

Configurazione di base

Nella configurazione di base, l'implementazione per la risoluzione DNS ibrida in un AWS ambiente multi-account utilizza AWS RAM la condivisione delle regole di inoltro di Resolver e delle associazioni di zone ospitate private per gestire le query DNS tra locale e risorse. AWS Questo metodo utilizza endpoint Route 53 Resolver centralizzati in un VPC di Shared Services connesso alla rete locale per gestire in modo efficiente la risoluzione DNS in entrata e in uscita.

- Per la risoluzione in uscita, le regole di inoltro Resolver vengono create nell'account Shared Services e quindi condivise con altri utenti tramite. Account AWS AWS RAM Questa condivisione è limitata agli account all'interno della stessa regione. Gli account di destinazione possono quindi associare queste regole alle proprie VPCs e abilitare le risorse di tali account VPCs per risolvere i nomi di dominio locali.

- Per la risoluzione in entrata, le zone ospitate private vengono create nell'account Shared Services e associate al VPC di Shared Services. Queste zone possono quindi essere associate VPCs ad altri account utilizzando l'API Route 53 o AWS Command Line Interface ()AWS CLI. AWS SDKs Le risorse associate VPCs possono quindi risolvere i record DNS definiti nelle zone ospitate private, creando una vista DNS unificata in tutto l'ambiente. AWS

Il diagramma seguente mostra i flussi di risoluzione DNS in questa configurazione di base.

Questa configurazione funziona bene quando lavori con un'infrastruttura DNS su scala limitata. Tuttavia, può diventare difficile da gestire man mano che l'ambiente cresce. Il sovraccarico operativo legato alla gestione VPCs individuale della zona ospitata privata e delle regole del Resolver aumenta notevolmente con la scalabilità. Inoltre, le quote di servizio come il limite di associazione di 300 VPC per zona ospitata privata possono diventare fattori vincolanti nelle implementazioni su larga scala. La configurazione avanzata risolve queste sfide.

Configurazione avanzata

I profili Route 53 offrono una soluzione semplificata per la gestione della risoluzione DNS in reti ibride su più reti. Account AWS Invece di gestire le zone private ospitate e le regole Resolver singolarmente, è possibile raggruppare le configurazioni DNS in un unico contenitore che può essere facilmente condiviso e applicato su più VPCs account in una regione. Questa configurazione mantiene l'architettura degli endpoint Resolver centralizzata in un VPC di Shared Services semplificando al contempo in modo significativo la gestione delle configurazioni DNS.

Il diagramma seguente mostra i flussi di risoluzione DNS in una configurazione avanzata.

I profili Route 53 consentono di raggruppare associazioni di zone ospitate private, regole di inoltro Resolver e regole firewall DNS in un'unica unità condivisibile. È possibile creare profili nell'account Shared Services e condividerli con gli account dei membri utilizzando. AWS RAM Quando un profilo viene condiviso e applicato a Target VPCs, tutte le associazioni e le configurazioni necessarie vengono gestite automaticamente dal servizio. Ciò riduce in modo significativo il sovraccarico operativo della gestione DNS e offre un'eccellente scalabilità per ambienti in crescita.

Automazione e scalabilità

Utilizza strumenti Infrastructure as Code (IaC) come CloudFormation o Terraform per effettuare il provisioning e gestire automaticamente gli endpoint, le regole, le zone ospitate private e i profili

di Route 53 Resolver. Integra la configurazione DNS con pipeline di integrazione continua e distribuzione continua (CI/CD) per coerenza, ripetibilità e aggiornamenti rapidi.

Tools (Strumenti)

Servizi AWS

- AWS Resource Access Manager (AWS RAM) consente di condividere in modo sicuro le risorse Account AWS per ridurre il sovraccarico operativo e fornire visibilità e verificabilità.
- Amazon Route 53 Resolverrisponde in modo ricorsivo alle richieste DNS provenienti dalle AWS risorse ed è disponibile per impostazione predefinita in tutte. VPCs Puoi creare endpoint Resolver e regole di inoltro condizionale per risolvere i namespace DNS tra il tuo data center locale e il tuo. VPCs
- La zona ospitata privata di Amazon Route 53 è un contenitore che contiene informazioni su come desideri che Route 53 risponda alle query DNS per un dominio e i relativi sottodomini.
- I profili Amazon Route 53 consentono di applicare e gestire configurazioni Route 53 relative al DNS su molte VPCs e diverse configurazioni Account AWS in modo semplificato.

Best practice

Questa sezione fornisce alcune delle migliori pratiche per ottimizzare Route 53 Resolver. Queste rappresentano un sottoinsieme delle best practice di Route 53. Per un elenco completo, consulta Best practice per Amazon Route 53.

Evita le configurazioni in loop con gli endpoint Resolver

- Progetta la tua architettura DNS per prevenire il routing ricorsivo pianificando attentamente le associazioni VPC. Quando un VPC ospita un endpoint in entrata, evita di associarlo a regole Resolver che potrebbero creare riferimenti circolari.
- Utilizzalo AWS RAM in modo strategico quando condividi le risorse DNS tra account per mantenere percorsi di routing puliti.

Per ulteriori informazioni, consulta Evita le configurazioni dei loop con gli endpoint Resolver nella documentazione di Route 53.

Scale gli endpoint di Resolver

- Per gli ambienti che richiedono un numero elevato di query al secondo (QPS), tieni presente che esiste un limite di 10.000 QPS per ENI in un endpoint. ENIs È possibile aggiungerne altri a un endpoint per scalare DNS QPS.
- CloudWatch Forniture `InboundQueryVolume` e `OutboundQueryVolume` parametri di Amazon (consulta la CloudWatch documentazione). Ti consigliamo di impostare regole di monitoraggio che ti avvisino se la soglia supera un determinato valore (ad esempio, l'80% di 10.000 QPS).
- Configura regole stateful security group per gli endpoint Resolver per evitare che i limiti di tracciamento delle connessioni causino la limitazione delle query DNS durante il traffico ad alto volume. Per ulteriori informazioni su come funziona il tracciamento delle connessioni nei gruppi di sicurezza di Amazon, consulta il monitoraggio delle connessioni dei gruppi di EC2 sicurezza Amazon nella EC2 documentazione di Amazon.

Per ulteriori informazioni, consulta Resolver Endpoint Scaling nella documentazione di Route 53.

Fornisci un'elevata disponibilità per gli endpoint Resolver

- Crea endpoint in entrata con indirizzi IP in almeno due zone di disponibilità per la ridondanza.
- Fornisci interfacce di rete aggiuntive per garantire la disponibilità durante la manutenzione o i picchi di traffico.

Per ulteriori informazioni, consulta Alta disponibilità per gli endpoint Resolver nella documentazione di Route 53.

Epiche

Implementa gli endpoint Route 53 Resolver

Operazione	Description	Competenze richieste
Implementa un endpoint in entrata.	Route 53 Resolver utilizza l'endpoint in entrata per ricevere query DNS dai resolver DNS locali. Per istruzioni, consulta Inoltro delle query DNS in entrata al tuo nella documentazione di Route 53. VPCs Prendi nota	Amministratore AWS, amministratore cloud

Operazione	Description	Competenze richieste
	dell'indirizzo IP dell'endpoint in entrata.	
Implementa un endpoint in uscita.	Route 53 Resolver utilizza l'endpoint in uscita per inviare query DNS ai resolver DNS locali. Per istruzioni, consulta Inoltro delle query DNS in uscita alla rete nella documentazione di Route 53. Prendi nota dell'ID dell'endpoint di output.	Amministratore AWS, amministratore cloud

Configura e condividi le zone private ospitate su Route 53

Operazione	Description	Competenze richieste
Crea una zona ospitata privata per un dominio ospitato su AWS.	Questa zona contiene i record DNS per le risorse in un dominio AWS ospitato (ad esempio,`prod.aws.example.com`) che devono essere risolti dall'ambiente locale. Per istruzioni, consulta Creazione di una zona ospitata privata nella documentazione di Route 53. Quando crei una zona ospitata privata, devi associare un VPC alla zona ospitata di proprietà dello stesso account. A tale scopo, selezionare il VPC di Shared Services.	Amministratore AWS, amministratore cloud

Operazione	Description	Competenze richieste
Configurazione di base: associa la zona ospitata privata VPCs ad altri account.	Se utilizzi la configurazione di base (vedi la sezione Architettura): Per consentire alle risorse dell'account membro di VPCs risolvere i record DNS in questa zona ospitata privata, devi VPCs associare la tua alla zona ospitata. È necessario autorizzare l'associazione e quindi crearla a livello di codice. Per istruzioni, consulta Associare un Amazon VPC e una zona ospitata privata che hai creato con Account AWS diversi nella documentazione di Route 53.	Amministratore AWS, amministratore cloud

Operazione	Description	Competenze richieste
Configurazione avanzata: configura e condividi i profili Route 53.	Se utilizzi una configurazione avanzata (consulta la sezione Architettura): 1. Crea un profilo Route 53 e associa ad esso le zone ospitate private pertinenti. Per istruzioni, consulta Creazione di profili Route 53 nella documentazione di Route 53. 2. AWS RAM Utilizzalo per condividere il profilo con gli account dei membri e quindi associare il profilo condiviso a target VPCs. Per istruzioni, consulta Condivisione di un profilo Route 53 e Associazione di un profilo Route 53 a VPCs nella documentazione di Route 53. **Note** A seconda della struttura dell'organizzazione e dei requisiti DNS, potrebbe essere necessario creare e gestire più profili per	Amministratore AWS, amministratore cloud

Operazione	Description	Competenze richieste
	account o carichi di lavoro diversi.	

Configura e condividi le regole di inoltro di Route 53 Resolver

Operazione	Description	Competenze richieste
Crea una regola di inoltro per un dominio ospitato in locale.	Questa regola indicherà a Route 53 Resolver di inoltrare qualsiasi query DNS per i domini locali (ad esempio) ai resolver DNS locali. `onprem.example.com` Per creare questa regola, sono necessari gli indirizzi IP dei resolver DNS locali e l'ID dell'endpoint in uscita. Per istruzioni, consulta Creazione di regole di inoltro nella documentazione di Route 53.	Amministratore AWS, amministratore cloud
Configurazione di base: condividi e associa la regola di inoltro VPCs a quella di altri account.	Se utilizzi la configurazione di base: Affinché la regola di inoltro abbia effetto, devi condivide rla e associarla VPCs a quella di altri account. Route 53 Resolver prende quindi in considerazione la regola quando risolve un dominio. Per istruzioni, consulta Condivisione delle regole del Resolver con altri utenti	Amministratore AWS, amministratore cloud

Operazione	Description	Competenze richieste
	Account AWS e utilizzo di regole condivise e Associazione delle regole di inoltro a un VPC nella documentazione di Route 53.	

Operazione	Description	Competenze richieste
Configurazione avanzata: configura e condividi i profili Route 53.	Se utilizzi una configurazione avanzata: 1. Se hai già creato un profilo Route 53 nei passaggi precedenti, puoi utilizzar e lo stesso profilo. In caso contrario, crea un profilo Route 53 e associa ad esso le regole di inoltro Resolver pertinenti. Per istruzioni, consulta Creazione di profili Route 53 nella documenta zione di Route 53. 2. AWS RAM Utilizzalo per condividere il profilo con gli account dei membri, quindi associa il profilo condiviso a target VPCs. Per istruzioni, consulta Condivisione di un profilo Route 53 e Associazi one di un profilo Route 53 a VPCs nella documenta zione di Route 53. **Note** A seconda della struttura dell'orga nizzazione e dei requisiti DNS, potrebbe essere necessario creare e gestire più profili per	Amministratore AWS, amministratore cloud

Operazione	Description	Competenze richieste
	account o carichi di lavoro diversi.	

Configura i resolver DNS locali per l'integrazione AWS

Operazione	Description	Competenze richieste
Configura l'inoltro condizionale nei resolver DNS locali.	Affinché le query DNS vengano inviate AWS dall'ambiente locale per la risoluzione, è necessario configurare l'inoltro condizion ale nei resolver DNS locali in modo che puntino all'indir izzo IP dell'endpoint in entrata. Ciò indica ai resolver DNS di inoltrare tutte le query DNS per il dominio ospitato (ad esempio, for) all'indirizzo IP dell'endpoint in entrata per la AWS risoluzione da parte di Route 53 Resolver. `prod.aws.example.com`	Amministratore di rete

Verifica la risoluzione DNS in un ambiente ibrido end-to-end

Operazione	Description	Competenze richieste
Verifica la risoluzione DNS dall' AWS ambiente locale.	Da un'istanza in un VPC a cui è associata la regola di inoltro, esegui una query DNS per un dominio ospitato in locale (ad	Amministratore di rete

Operazione	Description	Competenze richieste
	esempio, per). `db.onprem.example.com`	
Verifica la risoluzione DNS dall'ambiente locale a. AWS	Da un server locale, esegui la risoluzione DNS per un dominio AWS ospitato (ad esempio, per). `ec2.prod.aws.example.com`	Amministratore di rete

Risorse correlate

- Opzioni DNS cloud ibrido per Amazon VPC (white paper)AWS
- Utilizzo di zone ospitate private (documentazione Route 53)
- Guida introduttiva a Route 53 Resolver (documentazione Route 53)
- Semplifica la gestione DNS in un ambiente con più account con Route 53 Resolver (post sul blog)AWS
- Unifica la gestione DNS utilizzando profili Amazon Route 53 con più VPCs e Account AWS (AWS post di blog)
- Migrazione del tuo ambiente DNS multi-account ai profili Amazon Route 53 (AWS post del blog)
- Utilizzo dei profili Amazon Route 53 per AWS ambienti scalabili con più account (AWS post sul blog)

Verificare che i sistemi di bilanciamento del carico ELB richiedano la terminazione TLS

Priyanka Chaudhary, Amazon Web Services

Riepilogo

Sul cloud Amazon Web Services (AWS), Elastic Load Balancing (ELB) distribuisce automaticamente il traffico delle applicazioni in entrata su più destinazioni, come istanze Amazon Elastic Compute Cloud (Amazon EC2), contenitori, indirizzi IP e funzioni AWS Lambda. I sistemi di bilanciamento del carico utilizzano i listener per definire le porte e i protocolli utilizzati dal sistema di bilanciamento del carico per accettare il traffico dagli utenti. Gli Application Load Balancer prendono decisioni di routing a livello di applicazione e utilizzano i protocolli. HTTP/HTTPS I Classic Load Balancer prendono decisioni di routing a livello di trasporto, utilizzando i protocolli TCP o Secure Sockets Layer (SSL), o a livello di applicazione, utilizzando HTTP/HTTPS.

Questo modello fornisce un controllo di sicurezza che esamina diversi tipi di eventi per Application Load Balancers e Classic Load Balancers. Quando la funzione viene richiamata, AWS Lambda ispeziona l'evento e garantisce che il sistema di bilanciamento del carico sia conforme.

La funzione avvia un evento Amazon CloudWatch Events sulle seguenti chiamate API: CreateLoadBalancer, CreateLoadBalancerListeners, DeleteLoadBalancerListeners, CreateLoadBalancerPolicy, SetLoadBalancerPoliciesOfListener, CreateListenerDeleteListener, e ModifyListener. Quando l'evento rileva uno di questi APIs, chiama AWS Lambda, che esegue uno script Python. Lo script Python valuta se il listener contiene un certificato SSL e se la politica applicata utilizza Transport Layer Security (TLS). Se si determina che la politica SSL è diversa da TLS, la funzione invia una notifica Amazon Simple Notification Service (Amazon SNS) all'utente con le informazioni pertinenti.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo

Limitazioni

- Questo controllo di sicurezza non verifica la presenza di sistemi di bilanciamento del carico esistenti, a meno che non venga effettuato un aggiornamento dei listener del sistema di bilanciamento del carico.
- Questo controllo di sicurezza è regionale. Devi distribuirlo in ogni regione AWS che desideri monitorare.

Architecture

Architettura Target

Automazione e scalabilità

- Se utilizzi AWS Organizations, puoi utilizzare AWS Cloudformation StackSets per distribuire questo modello in più account che desideri monitorare.

Tools (Strumenti)

Servizi AWS

- AWS CloudFormation: AWS ti CloudFormation aiuta a modellare e configurare le tue risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle per tutto il loro ciclo di vita. Puoi utilizzare un modello per descrivere le tue risorse e le loro dipendenze e lanciarle e configurarle insieme come uno stack, invece di gestire le risorse singolarmente.
- Amazon CloudWatch Events: Amazon CloudWatch Events offre un flusso quasi in tempo reale di eventi di sistema che descrivono i cambiamenti nelle risorse AWS.
- AWS Lambda: AWS Lambda è un servizio di elaborazione che supporta l'esecuzione di codice senza effettuare il provisioning o la gestione di server.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti altamente scalabile che può essere utilizzato per un'ampia gamma di soluzioni di storage, tra cui siti Web, applicazioni mobili, backup e data lake.
- Amazon SNS — Amazon Simple Notification Service (Amazon SNS) coordina e gestisce la consegna o l'invio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail. I sottoscrittori ricevono tutti gli stessi messaggi pubblicati sugli argomenti ai quali sono hanno effettuato la sottoscrizione.

Codice

Questo modello include i seguenti allegati:

- `ELBRequirestlstermination.zip`— Il codice Lambda per il controllo di sicurezza.
- `ELBRequirestlstermination.yml`— Il CloudFormation modello che configura l'evento e la funzione Lambda.

Epiche

Configura il bucket S3

Operazione	Description	Competenze richieste
Definisci il bucket S3.	Sulla console Amazon S3, scegli o crea un bucket S3 per ospitare il file.zip con codice Lambda. Questo bucket S3 deve trovarsi nella stessa regione AWS del sistema di bilanciamento del carico che desideri valutare. Il nome di un bucket S3 è unico a livello globale e lo spazio dei nomi è condiviso da tutti gli account AWS. Il nome del bucket S3 non può includere barre iniziali.	Architetto del cloud
Carica il codice Lambda.	Carica il codice Lambda (`ELBRequirestlstermination.zip` file) fornito nella sezione Allegati nel bucket S3.	Architetto del cloud

Implementa il modello CloudFormation

Operazione	Description	Competenze richieste
Avvia il CloudFormation modello AWS.	Apri la CloudFormation console AWS nella stessa regione AWS del bucket S3 e distribuisci il modello allegato. `ELBRequirestlstermination.yml` Per ulteriori informazioni sulla distribuzione di CloudFormation modelli AWS, consulta Creazione di uno stack sulla CloudFormation console AWS nella CloudFormation documentazione.	Architetto del cloud
Completa i parametri nel modello.	Quando avvii il modello, ti verranno richieste le seguenti informazioni: • Bucket S3: specifica il bucket che hai creato o selezionato nella prima epic. Qui è dove hai caricato il codice Lambda allegato (`ELBRequirestlstermination.zip` file). • Chiave S3: specifica la posizione del file.zip Lambda nel bucket S3 (ad esempio o). `ELBRequirestlstermination.zip controls/ELBRequirestlsterm`	Architetto del cloud

Operazione	Description	Competenze richieste
	`ination.zip` Non includere le barre iniziali. • E-mail di notifica: fornisci un indirizzo e-mail attivo a cui desideri ricevere le notifiche di Amazon SNS. • Livello di registrazione Lambda: specifica il livello e la frequenza di registrazione per la funzione Lambda. Utilizzate Info per registrar e messaggi informativi dettagliati sullo stato di avanzamento, Errore per gli eventi di errore che potrebbero comunque consentire la continuazione della distribuzione e Avviso per situazioni potenzial mente dannose.	

Confermare la sottoscrizione

Operazione	Description	Competenze richieste
Confermare la sottoscrizione.	Quando il CloudFormation modello viene distribuito correttamente, invia un'e-mail di iscrizione all'indirizzo e-mail fornito. È necessario confermare questa sottoscri zione e-mail per iniziare a ricevere notifiche di violazione.	Architetto del cloud

Risorse correlate

- Creazione di uno stack sulla CloudFormation console AWS (CloudFormation documentazione AWS)
- Cos'è AWS Lambda? (documentazione AWS Lambda)
- Cos'è un Classic Load Balancer? (documentazione ELB)
- Cos’è un Application Load Balancer? (documentazione ELB)

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Visualizza i log e i parametri di AWS Network Firewall utilizzando Splunk

Ivo Pinto, Amazon Web Services

Riepilogo

Molte organizzazioni utilizzano Splunk Enterprise come strumento centralizzato di aggregazione e visualizzazione per log e metriche provenienti da diverse fonti. Questo modello ti aiuta a configurare Splunk per recuperare i log e i parametri di AWS Network Firewall da CloudWatch Amazon Logs utilizzando il componente aggiuntivo Splunk per AWS.

A tal fine, crei un ruolo AWS Identity and Access Management (IAM) di sola lettura. Splunk Add-On for AWS utilizza questo ruolo per accedere. CloudWatch Puoi configurare il componente aggiuntivo Splunk per AWS da cui recuperare metriche e log. CloudWatch Infine, crei visualizzazioni in Splunk a partire dai dati e dalle metriche dei log recuperati.

Prerequisiti e limitazioni

Prerequisiti

- Un account Splunk
- Un'istanza Splunk Enterprise, versione 8.2.2 o successiva
- Un account AWS attivo
- Network Firewall, configurato e configurato per inviare log a CloudWatch Logs

Limitazioni

- Splunk Enterprise deve essere distribuito come cluster di istanze Amazon Elastic Compute Cloud (Amazon EC2) nel cloud AWS.
- La raccolta di dati utilizzando un ruolo IAM scoperto automaticamente per Amazon EC2 non è supportata nelle regioni AWS Cina.

Architecture

Il diagramma illustra quanto segue:

1. Network Firewall pubblica i log in Logs. CloudWatch

2. Splunk Enterprise recupera metriche e log da. CloudWatch

Per compilare metriche e log di esempio in questa architettura, un carico di lavoro genera traffico che attraversa l'endpoint Network Firewall per andare a Internet. Questo risultato è ottenuto mediante l'uso di tabelle di routing. Sebbene questo modello utilizzi una singola istanza Amazon EC2 come carico di lavoro, può essere applicato a qualsiasi architettura purché Network Firewall sia configurato per inviare log a Logs. CloudWatch

Questa architettura utilizza anche un'istanza Splunk Enterprise in un altro cloud privato virtuale (VPC). Tuttavia, l'istanza Splunk può trovarsi in un'altra posizione, ad esempio nello stesso VPC del carico di lavoro, purché possa raggiungere il. CloudWatch APIs

Tools (Strumenti)

Servizi AWS

- Amazon CloudWatch Logs ti aiuta a centralizzare i log di tutti i tuoi sistemi, applicazioni e servizi AWS in modo da poterli monitorare e archiviare in modo sicuro.
- Amazon Elastic Compute Cloud (Amazon EC2) Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- AWS Network Firewall è un firewall di rete a stato gestito e un servizio di rilevamento e prevenzione delle intrusioni VPCs nel cloud AWS.

Altri strumenti

- Splunk ti aiuta a monitorare, visualizzare e analizzare i dati di registro.

Epiche

Creazione di un ruolo IAM

Operazione	Description	Competenze richieste
Creare la policy IAM.	Segui le istruzioni in Creazione di policy using the JSON editor per creare la	Amministratore AWS

Operazione	Description	Competenze richieste
	policy IAM che garantisce l'accesso in sola lettura ai dati e alle metriche dei CloudWatch Logs. CloudWatch Incollare la seguente policy nell'editor JSON. `{ "Statement": [ { "Action": [ "cloudwatch:List*", "cloudwatch:Get*", "network-firewall:List*", "logs:Describe*", "logs:Get*", "logs:List*", "logs:StartQuery", "logs:StopQuery", "logs:TestMetricFilter", "logs:FilterLogEvents", "network-firewall:Describe*" ], "Effect": "Allow",`	

Operazione	Description	Competenze richieste
	<pre>"Resource": "*" }], "Version": "2012-10-17" }</pre>	
Crea un nuovo ruolo IAM.	Segui le istruzioni in Creazione di un ruolo per delegare le autorizzazioni a un servizio AWS per creare il ruolo IAM a cui il componente aggiuntivo Splunk per AWS utilizza per accedere. CloudWatch Per le politiche di autorizzazione, scegli la politica che hai creato in precedenza.	Amministratore AWS
Assegna il ruolo IAM alle istanze EC2 nel cluster Splunk.	1. Apri la console Amazon EC2 all'indirizzo https://console.aws.amazon.com/ec2/. 2. Nel riquadro di navigazione, scegliere Instances (Istanze). 3. Seleziona le istanze EC2 nel cluster Splunk. 4. Scegli Azioni, Sicurezza e poi Modifica il ruolo IAM. 5. Seleziona il ruolo IAM che hai creato in precedenza, quindi scegli Salva.	Amministratore AWS

Installa il componente aggiuntivo Splunk per AWS

Operazione	Description	Competenze richieste
Installa il componente aggiuntivo .	1. Nella dashboard di Splunk, accedi a Splunk Apps. 2. Cerca il componente aggiuntivo Splunk per Amazon Web Services. 3. Scegli Installa. 4. Fornisci le tue credenziali Splunk.	Amministratore Splunk
Configura le credenziali AWS.	1. Nella dashboard Splunk, accedi al componente aggiuntivo Splunk per AWS. 2. Scegliere Configuration (Configurazione). 3. Nella colonna Autodisco vered IAM Role, seleziona il ruolo IAM che hai creato in precedenza. Per ulteriori informazioni, consulta Trova un ruolo IAM all'interno dell'istanza della piattaforma Splunk nella documentazione Splunk.	Amministratore Splunk

Configura l'accesso Splunk a CloudWatch

Operazione	Description	Competenze richieste
Configurare il recupero dei log del Network Firewall dai registri. CloudWatch	1. Nella dashboard Splunk, accedi al componente aggiuntivo Splunk per AWS. 2. Scegli Input. 3. Scegli Crea nuovo input. 4. Nell'elenco, scegli Tipo di dati personalizzato, quindi scegli CloudWatch Registri. 5. Fornisci il nome, l'account AWS, la regione AWS e il gruppo di log per i log del Network Firewall. 6. Scegli Save (Salva). Per impostazione predefini ta, Splunk recupera i dati di registro ogni 10 minuti. Questo è un parametro configurabile in Impostazioni avanzate. Per ulteriori informazioni, consulta Configurare un input di CloudWatch log utilizzando Splunk Web nella documenta zione di Splunk.	Amministratore Splunk
Configura il recupero delle metriche del Network Firewall da. CloudWatch	1. Nella dashboard Splunk, accedi al componente aggiuntivo Splunk per AWS. 2. Scegli Input. 3. Scegli Crea nuovo input.	Amministratore Splunk

Operazione	Description	Competenze richieste
	4. Nell'elenco, scegli CloudWatch. 5. Fornisci il nome, l'account AWS e la regione AWS per i parametri del Network Firewall. 6. Accanto a Metric Configuration, scegli Modifica in modalità avanzata. 7. (Facoltativo) Eliminate tutti i namespace preconfigurati. 8. Scegli Aggiungi namespace, quindi denominalo AWS/NetworkFirewall 9. In Dimension Value, aggiungi quanto segue. `[{"AvailabilityZone":[".*"],"Engine":[".*"],"FirewallName":[".*"]}]` 10. Per Metriche, scegli Tutto. 11. Per Statistiche metriche, scegliete Somma. 12. Scegli OK. 13. Scegli Save (Salva). Per impostazione predefinita, Splunk recupera i dati metrici ogni 5 minuti. Questo è un parametro configurabile in Impostazioni avanzate. Per ulteriori informazioni, consulta	

Operazione	Description	Competenze richieste
	Configurare un CloudWatch input utilizzando Splunk Web nella documentazione di Splunk.	

Crea visualizzazioni Splunk utilizzando le query

Operazione	Description	Competenze richieste
Visualizza i principali indirizzi IP di origine.	1. Nella dashboard di Splunk, accedi a Search & Reporting. 2. Nella casella Inserisci la ricerca qui, inserisci quanto segue. `sourcetype="aws:cloudwatchlogs" | top event.src_ip` Questa query visualizza una tabella degli indirizzi IP di origine con il maggior traffico, in ordine decrescente. 3. Per una rappresentazione grafica, scegli Visualizzazione.	Amministratore Splunk
Visualizza le statistiche sui pacchetti.	1. Nella dashboard di Splunk, accedi a Search & Reporting.	Amministratore Splunk

Operazione	Description	Competenze richieste
	2. Nella casella Inserisci la ricerca qui, inserisci quanto segue. `sourcetype="aws:cloudwatch"| timechart sum(Sum) by metric_name` Questa query visualizza una tabella delle metriche `DroppedPackets` e `ReceivedPackets` al minuto. `PassedPackets` 3. Per una rappresentazione grafica, scegliete Visualizzazione.	

Operazione	Description	Competenze richieste
Visualizza le porte di origine più utilizzate.	1. Nella dashboard di Splunk, accedi a Search & Reporting. 2. Nella casella Inserisci la ricerca qui, inserisci quanto segue. `sourcetype="aws:cloudwatchlogs" | top event.dest_port` Questa query visualizza una tabella delle porte di origine con il maggior traffico, in ordine decrescente. 3. Per una rappresentazione grafica, scegli Visualizzazione.	Amministratore Splunk

Risorse correlate

Documentazione AWS

- Creazione di un ruolo per delegare le autorizzazioni a un servizio AWS (documentazione IAM)
- Creazione di politiche IAM (documentazione IAM)
- Registrazione e monitoraggio in AWS Network Firewall (documentazione Network Firewall)
- Configurazioni della tabella di routing per AWS Network Firewall (documentazione Network Firewall)

Post sul blog di AWS

- Modelli di implementazione di AWS Network Firewall

AWS Marketplace

- Splunk Enterprise Amazon Machine Image (AMI)

Altri modelli

- Accedi a un host bastion utilizzando Session Manager e Amazon EC2 Instance Connect
- Accedi alle applicazioni container in modo privato su Amazon ECS utilizzando AWS Fargate, PrivateLink AWS e un Network Load Balancer
- Accedi alle applicazioni container in modo privato su Amazon ECS utilizzando AWS PrivateLink e un Network Load Balancer
- Centralizza la risoluzione DNS utilizzando AWS Managed Microsoft AD e localmente Microsoft Active Directory
- Crea un portale per micro-frontend utilizzando Angular e AWS Amplify Module Federation
- Implementa un'API Amazon API Gateway su un sito Web interno utilizzando endpoint privati e un Application Load Balancer
- Implementa controlli di accesso basati su attributi investigativi per sottoreti pubbliche utilizzando AWS Config
- Implementa controlli di accesso preventivi basati sugli attributi per le sottoreti pubbliche
- Abilita connessioni crittografate per le istanze DB PostgreSQL in Amazon RDS
- Estendi VRFs ad AWS utilizzando AWS Transit Gateway Connect
- Esegui la migrazione di un carico di lavoro F5 BIG-IP su F5 BIG-IP VE su Cloud AWS
- Esegui la migrazione dei controller di ingresso NGINX quando abiliti la modalità automatica di Amazon EKS
- Conserva lo spazio IP instradabile nei progetti VPC multi-account per sottoreti non destinate ai carichi di lavoro
- Impedisci l'accesso a Internet a livello di account utilizzando una politica di controllo del servizio
- Invia avvisi da AWS Network Firewall a un canale Slack
- Distribuisci contenuti statici in un bucket Amazon S3 tramite un VPC utilizzando Amazon CloudFront
- Configura il disaster recovery per Oracle JD Edwards con EnterpriseOne AWS Elastic Disaster Recovery
- Usa le query BMC Discovery per estrarre i dati di migrazione per la pianificazione della migrazione
- Usa Network Firewall per acquisire i nomi di dominio DNS dall'Indicazione del nome del server per il traffico in uscita

Distribuzione di contenuti

Argomenti

- Invia AWS WAF i log a Splunk utilizzando AWS Firewall Manager Amazon Data Firehose
- Distribuisci contenuti statici in un bucket Amazon S3 tramite un VPC utilizzando Amazon CloudFront
- Altri modelli

Invia AWS WAF i log a Splunk utilizzando AWS Firewall Manager Amazon Data Firehose

Michael Friedenthal, Aman Kaur Gandhi e JJ Johnson, Amazon Web Services

Riepilogo

Storicamente, esistevano due modi per spostare i dati in Splunk: un'architettura push o una pull. Un'architettura pull offre garanzie di consegna dei dati attraverso nuovi tentativi, ma richiede risorse dedicate in Splunk per raccogliere i dati. Le architetture pull di solito non sono in tempo reale a causa del polling. Un'architettura push in genere ha una latenza inferiore, è più scalabile e riduce la complessità e i costi operativi. Tuttavia, non garantisce la consegna e in genere richiede agenti.

L'integrazione di Splunk con Amazon Data Firehose fornisce dati di streaming in tempo reale a Splunk tramite un HTTP Event Collector (HEC). Questa integrazione offre i vantaggi delle architetture push e pull: garantisce la consegna dei dati tramite nuovi tentativi, è quasi in tempo reale, è a bassa latenza e bassa complessità. L'HEC invia dati in modo rapido ed efficiente tramite HTTP o HTTPS direttamente a Splunk. HECs sono basati su token, il che elimina la necessità di codificare le credenziali in un'applicazione o nei file di supporto.

In base a una AWS Firewall Manager policy, puoi configurare la registrazione per tutto il traffico ACL AWS WAF Web in tutti i tuoi account e quindi utilizzare un flusso di distribuzione Firehose per inviare i dati di registro a Splunk per il monitoraggio, la visualizzazione e l'analisi. Questa soluzione offre i seguenti vantaggi:

- Gestione e registrazione centralizzate del traffico ACL AWS WAF Web in tutti gli account
- Integrazione Splunk con un unico Account AWS
- Scalabilità
- Distribuzione quasi in tempo reale dei dati di registro
- Ottimizzazione dei costi attraverso l'uso di una soluzione serverless, in modo da non dover pagare per le risorse inutilizzate.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS che fa parte di un'organizzazione in. AWS Organizations

- È necessario disporre delle seguenti autorizzazioni per abilitare la registrazione con Firehose:
  - `iam:CreateServiceLinkedRole`
  - `firehose:ListDeliveryStreams`
  - `wafv2:PutLoggingConfiguration`
- AWS WAF e il relativo sito web ACLs deve essere configurato. Per istruzioni, vedi Guida introduttiva a AWS WAF.
- AWS Firewall Manager deve essere configurato. Per istruzioni, consulta i AWS Firewall Manager prerequisiti.
- Le politiche di sicurezza di Firewall Manager per AWS WAF devono essere configurate. Per istruzioni, vedere Guida introduttiva alle AWS Firewall ManagerAWS WAF politiche.
- Splunk deve essere configurato con un endpoint HTTP pubblico raggiungibile da Firehose.

Limitazioni

- Account AWS Devono essere gestiti in un'unica organizzazione in. AWS Organizations
- L'ACL Web deve trovarsi nella stessa regione del flusso di consegna. Se stai acquisendo log per Amazon CloudFront, crea il flusso di consegna Firehose nella regione Stati Uniti orientali (Virginia settentrionale),. `us-east-1`
- Il componente aggiuntivo Splunk per Firehose è disponibile per le distribuzioni Splunk Cloud a pagamento, le implementazioni distribuite di Splunk Enterprise e le implementazioni Splunk Enterprise a istanza singola. Questo componente aggiuntivo non è supportato per le distribuzioni di prova gratuite di Splunk Cloud.

Architecture

Stack tecnologico Target

- Firewall Manager
- Firehose
- Amazon Simple Storage Service (Amazon S3)
- AWS WAF
- Splunk

Architettura di destinazione

L'immagine seguente mostra come utilizzare Firewall Manager per registrare centralmente tutti i AWS WAF dati e inviarli a Splunk tramite Firehose.

1. Il AWS WAF Web ACLs invia i dati di registro del firewall a Firewall Manager.
2. Firewall Manager invia i dati di registro a Firehose.
3. Il flusso di distribuzione Firehose inoltra i dati di registro a Splunk e a un bucket S3. Il bucket S3 funge da backup in caso di errore nel flusso di distribuzione di Firehose.

Automazione e scalabilità

Questa soluzione è progettata per scalare e adattarsi a tutto il AWS WAF Web ALCs all'interno dell'organizzazione. È possibile configurare tutto il Web ACLs per utilizzare la stessa istanza di Firehose. Tuttavia, se desideri configurare e utilizzare più istanze di Firehose, puoi farlo.

Tools (Strumenti)

Servizi AWS

- AWS Firewall Managerè un servizio di gestione della sicurezza che consente di configurare e gestire centralmente le regole del firewall tra gli account e le applicazioni in. AWS Organizations
- Amazon Data Firehose ti aiuta a distribuire dati di streaming in tempo reale ad altri Servizi AWS endpoint HTTP personalizzati e endpoint HTTP di proprietà di fornitori di servizi terzi supportati, come Splunk.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS WAFè un firewall per applicazioni Web che consente di monitorare le richieste HTTP e HTTPS inoltrate alle risorse protette delle applicazioni Web.

Altri strumenti

- Splunk ti aiuta a monitorare, visualizzare e analizzare i dati di registro.

Epiche

Configura Splunk

Operazione	Description	Competenze richieste
Installa l'app Splunk per AWS.	1. Accedi al tuo spedizioniere pesante Splunk. L'URL predefinito è. `http://<IP address>:8000` 2. Nella barra di navigazione a sinistra, accanto a App, scegli il pulsante a forma di ingranaggio. 3. Scegli Sfoglia altre app. 4. Cercare AWS. 5. Per l'app Splunk per AWS, scegli Installa. 6. Inserisci le tue credenziali di accesso a Splunk.com, accetta i termini e le condizioni, quindi scegli Accedi e installa. 7. Seleziona Fatto.	Amministratore della sicurezza, amministratore Splunk
Installa il componente aggiuntivo per. AWS WAF	Ripeti le istruzioni precedenti per installare il componente aggiuntivo AWS Web Application Firewall per Splunk.	Amministratore della sicurezza, amministratore Splunk
Installa e configura il componente aggiuntivo Splunk per Firehose.	1. Installa e configura il componente aggiuntivo Splunk per Firehose. Come parte dell'installazione e della configurazione, se necessario per la	Amministratore della sicurezza, amministratore Splunk

Operazione	Description	Competenze richieste
	tua piattaforma Splunk, configuri un HTTP Event Collector e prepari l'infrast ruttura per inviare i dati di registro ai tuoi indicizza tori. Consulta le istruzioni che corrispondono alla tua implementazione Splunk: • Implementazione di Splunk Cloud (document azione Splunk) • Implementazione distribui ta di Splunk Enterprise (documentazione Splunk) • Implementazione Splunk Enterprise a istanza singola (documentazione Splunk) **Important** Interrompi questa procedura dopo aver installato e configurato il componente aggiuntivo Splunk. Non procedere con le istruzion i per configurare Firehose per l'invio di dati alla piattafor ma Splunk.	

Operazione	Description	Competenze richieste
	2. Prendi nota del token HTTP Event Collector e dell'endpoint HTTP. Questo valore ti servirà in seguito, quando configurerai il flusso di distribuzione.	

Creare il flusso di distribuzione di Firehose

Operazione	Description	Competenze richieste
Concedi a Firehose l'accesso a una destinazione Splunk.	Configura la politica di accesso che consente a Firehose di accedere a una destinazione Splunk e di eseguire il backup dei dati di registro su un bucket S3. Per ulteriori informazioni, consulta Concedere a Firehose l'accesso a una destinazione Splunk.	Amministratore della sicurezza
Crea un flusso di distribuzione Firehose.	Nello stesso account ACLs per cui gestisci il Web AWS WAF, crea un flusso di distribuzione in Firehose. Quando crei un flusso di distribuzione, devi avere un ruolo IAM. Firehose assume quel ruolo IAM e ottiene l'accesso al bucket S3 specificato. Per istruzioni, consulta Creazione di un flusso di distribuzione. Tenere presente quanto segue:	Amministratore della sicurezza

Operazione	Description	Competenze richieste
	• Il nome del flusso di consegna deve iniziare con `aws-waf-logs-` . • Come sorgente, scegliete Direct PUT. • Per la modalità di backup S3, scegli Backup di tutti gli eventi, quindi scegli un bucket esistente o creane uno nuovo. • Per la destinazione, segui le istruzioni in Scegli Splunk per la tua destinazione nella documentazione di Firehose. Per informazioni sui valori per gli endpoint e i tipi di endpoint Splunk, consulta Configure Amazon Data Firehose nella documentazione di Splunk. Ripeti questo processo per ogni token configurato nel raccoglitore di eventi HTTP.	
Testa il flusso di distribuzione.	Testa il flusso di consegna per verificare che sia configurato correttamente. Per istruzioni, consulta Test using Splunk come destinazione nella documentazione di Firehose.	Amministratore della sicurezza

Configurare Firewall Manager per registrare i dati

Operazione	Description	Competenze richieste
Configurare le politiche del Firewall Manager.	Le policy di Firewall Manager devono essere configura te per abilitare la registraz ione e inoltrare i log al flusso di distribuzione Firehose corretto. Per ulteriori informazi oni e istruzioni, vedere Configurazione della registraz ione per una policy. AWS WAF	Amministratore della sicurezza

Risorse correlate

AWS resources

- Registrazione del traffico ACL web (documentazione)AWS WAF
- Configurazione della registrazione per una AWS WAF politica (documentazione)AWS WAF
- Tutorial: invio di log di flusso VPC a Splunk utilizzando Amazon Data Firehose (documentazione Firehose)
- In che modo posso inviare i log di flusso VPC a Splunk utilizzando Amazon Data Firehose? (Centro di conoscenza)AWS
- Potenzia l'inserimento dei dati in Splunk utilizzando Amazon Data Firehose (post sul blog)AWS

Documentazione Splunk

- Componente aggiuntivo Splunk per Amazon Data Firehose

Distribuisci contenuti statici in un bucket Amazon S3 tramite un VPC utilizzando Amazon CloudFront

Ángel Emmanuel Hernández Cebrian, Amazon Web Services

Riepilogo

Quando offri contenuti statici ospitati su Amazon Web Services (AWS), l'approccio consigliato consiste nell'utilizzare un bucket Amazon Simple Storage Service (S3) come origine e utilizzare Amazon CloudFront per distribuire il contenuto. Questa soluzione offre due vantaggi principali: la comodità di memorizzare nella cache i contenuti statici nelle postazioni periferiche e la possibilità di definire liste di controllo degli accessi Web (web ACLs) per la CloudFront distribuzione, che aiutano a proteggere le richieste ai contenuti con una configurazione e un sovraccarico amministrativo minimi.

Tuttavia, esiste una limitazione architetturale comune all'approccio standard consigliato. In alcuni ambienti, si desidera che le appliance firewall virtuali siano distribuite in un cloud privato virtuale (VPC) per ispezionare tutti i contenuti, inclusi i contenuti statici. L'approccio standard non indirizza il traffico attraverso il VPC per l'ispezione. Questo modello fornisce una soluzione architettonica alternativa. Continui a utilizzare una CloudFront distribuzione per fornire contenuti statici in un bucket S3, ma il traffico viene instradato attraverso il VPC utilizzando un Application Load Balancer. Una funzione AWS Lambda recupera e restituisce quindi il contenuto dal bucket S3.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Contenuto statico del sito Web ospitato in un bucket S3.

Limitazioni

- Le risorse in questo modello devono trovarsi in una singola regione AWS, ma possono essere fornite in diversi account AWS.
- I limiti si applicano alla dimensione massima di richiesta e risposta che la funzione Lambda può ricevere e inviare, rispettivamente. Per ulteriori informazioni, consulta Limiti nelle funzioni Lambda come destinazioni (documentazione Elastic Load Balancing).
- È importante trovare un buon equilibrio tra prestazioni, scalabilità, sicurezza ed economicità quando si utilizza questo approccio. Nonostante l'elevata scalabilità di Lambda, se il numero di

chiamate Lambda simultanee supera la quota massima, alcune richieste vengono limitate. Per ulteriori informazioni, consulta Quote Lambda (documentazione Lambda). È inoltre necessario considerare i prezzi quando si utilizza Lambda. Per ridurre al minimo le chiamate Lambda, assicurati di definire correttamente la cache per la distribuzione. CloudFront Per ulteriori informazioni, consulta Ottimizzazione della memorizzazione nella cache e della disponibilità (documentazione). CloudFront

Architecture

Stack tecnologico Target

- CloudFront
- Amazon Virtual Private Cloud (Amazon VPC) (Amazon VPC)
- Application Load Balancer
- Lambda
- Simple Storage Service (Amazon S3)

Architettura di destinazione

L'immagine seguente mostra l'architettura consigliata quando è necessario utilizzare CloudFront per servire contenuti statici da un bucket S3 tramite un VPC.

1. Il client richiede l'URL di CloudFront distribuzione per inserire un particolare file del sito Web nel bucket S3.
2. CloudFront invia la richiesta ad AWS WAF. AWS WAF filtra la richiesta utilizzando il Web ACLs applicato alla CloudFront distribuzione. Se la richiesta viene ritenuta valida, il flusso continua. Se la richiesta viene ritenuta non valida, il client riceve un errore 403.
3. CloudFront controlla la sua cache interna. Se esiste una chiave valida corrispondente alla richiesta in entrata, il valore associato viene rispedito al client come risposta. In caso contrario, il flusso continua.
4. CloudFront inoltra la richiesta all'URL dell'Application Load Balancer specificato.
5. L'Application Load Balancer ha un listener associato a un gruppo target basato su una funzione Lambda. L'Application Load Balancer richiama la funzione Lambda.

6. La funzione Lambda si connette al bucket S3, esegue un'`GetObject`operazione su di esso e restituisce il contenuto come risposta.

Automazione e scalabilità

Per automatizzare la distribuzione di contenuti statici utilizzando questo approccio, crea CI/CD pipeline per aggiornare i bucket Amazon S3 che ospitano i siti Web.

La funzione Lambda si ridimensiona automaticamente per gestire le richieste concorrenti, entro le quote e le limitazioni del servizio. Per ulteriori informazioni, consulta Scalabilità delle funzioni Lambda e quote Lambda (documentazione Lambda). Per gli altri servizi e funzionalità AWS, come CloudFront Application Load Balancer, AWS li ridimensiona automaticamente.

Tools (Strumenti)

- Amazon CloudFront accelera la distribuzione dei tuoi contenuti web distribuendoli attraverso una rete mondiale di data center, che riduce la latenza e migliora le prestazioni.
- Elastic Load Balancing (ELB) distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni. In questo modello, si utilizza un Application Load Balancer fornito tramite Elastic Load Balancing per indirizzare il traffico verso la funzione Lambda.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito. Questa rete virtuale è simile a una rete tradizionale che gestiresti nel tuo data center, con i vantaggi dell'utilizzo dell'infrastruttura scalabile di AWS.

Epiche

Utilizzalo CloudFront per servire contenuti statici da Amazon S3 tramite un VPC

Operazione	Description	Competenze richieste
Crea un VPC.	Crea un VPC per ospitare le risorse distribuite in questo	Architetto del cloud

Operazione	Description	Competenze richieste
	modello, come l'Application Load Balancer e la funzione Lambda. Per istruzioni, consulta Creare un VPC (documentazione Amazon VPC).	
Crea un ACL web AWS WAF.	Crea un ACL web AWS WAF. Più avanti in questo schema, applicherai questo ACL web alla distribuzione. CloudFron t Per istruzioni, consulta Creazione di un ACL web (documentazione AWS WAF).	Architetto del cloud
Creazione della funzione Lambda	Crea la funzione Lambda che serve il contenuto statico ospitato nel bucket S3 come sito Web. Utilizza il codice fornito nella sezione Informazi oni aggiuntive di questo modello. Personalizza il codice per identificare il bucket S3 di destinazione.	Informazioni generali su AWS
Carica la funzione Lambda.	Immettete il seguente comando per caricare il codice della funzione Lambda in un archivio di file.zip in Lambda. `aws lambda update-fu nction-code \ --function-name \ --zip-file fileb://l ambda-alb-s3-websi te.zip`	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Crea un Application Load Balancer.	Crea un Application Load Balancer con accesso a Internet che punti alla funzione Lambda. Per istruzioni, consulta Creare un gruppo target per la funzione Lambda (documentazione Elastic Load Balancing). Per una configura zione ad alta disponibilità, crea l'Application Load Balancer e collegalo a sottoreti private in diverse zone di disponibilità.	Architetto del cloud

Operazione	Description	Competenze richieste
Crea una CloudFront distribuzione.	Crea una CloudFront distribuzione che punti all'Application Load Balancer che hai creato. 1. Accedi alla Console di gestione AWS e apri la CloudFront console su https://console.aws.amazon.com/cloudfront/v3/home. 2. Scegliere Create Distribution (Crea distribuzione). 3. Nella prima pagina della procedura guidata Create Distribution (Crea distribuzione), nella sezione Web, scegli Get Started (Inizia). 4. Specificate le impostazioni per la vostra distribuzione. Per ulteriori informazioni, consulta Valori da specificare durante la creazione o l'aggiornamento di una distribuzione. Tenere presente quanto segue: a. Imposta Application Load Balancer come origine. b. Nelle impostazioni di distribuzione, scegli il Web esistente ACLs che desideri applicare tramite AWS WAF. Per ulteriori informazioni, consulta l'ACL web AWS WAF. 5. Salvare le modifiche.	Architetto del cloud

Operazione	Description	Competenze richieste
	6. Dopo aver CloudFront creato la distribuzione, il valore della colonna Status relativa InProgressalla distribuzione cambia da Deployed. Se scegli di abilitare la distribuzione, sarà pronta per elaborare le richieste dopo viene attivato lo stato Deployed (Distribu ito).	

Risorse correlate

Documentazione AWS

- Ottimizzazione della memorizzazione nella cache e della disponibilità (documentazione) CloudFront
- Lambda funge da obiettivi (documentazione Elastic Load Balancing)
- Quote Lambda (documentazione Lambda)

Siti Web di servizi AWS

- Application Load Balancer
- Lambda
- CloudFront
- Amazon S3
- AWS WAF
- Amazon VPC

Informazioni aggiuntive

Codice

Il seguente esempio di funzione Lambda è scritto in Node.js. Questa funzione Lambda funge da server Web che esegue un'`GetObject`operazione su un bucket S3 che contiene le risorse del sito Web.

```
/**

 * This is an AWS Lambda function created for demonstration purposes.

 * It retrieves static assets from a defined Amazon S3 bucket.

 * To make the content available through a URL, use an Application Load Balancer with a
 Lambda integration.
 *
 * Set the S3_BUCKET environment variable in the Lambda function definition.
 */

var AWS = require('aws-sdk');

exports.handler = function(event, context, callback) {

    var bucket = process.env.S3_BUCKET;
    var key = event.path.replace('/', '');

    if (key == '') {
        key = 'index.html';
    }

    // Fetch from S3
    var s3 = new AWS.S3();
    return s3.getObject({Bucket: bucket, Key: key},
       function(err, data) {

            if (err) {
                return err;
            }

            var isBase64Encoded = false;
            var encoding = 'utf8';

            if (data.ContentType.indexOf('image/') > -1) {
                isBase64Encoded = true;
                encoding = 'base64'
            }
```

```
            var resp = {
                statusCode: 200,
                headers: {
                    'Content-Type': data.ContentType,
                },
                body: new Buffer(data.Body).toString(encoding),
                isBase64Encoded: isBase64Encoded
            };

            callback(null, resp);
        }
    );
};
```

Altri modelli

- Controlla una CloudFront distribuzione Amazon per la registrazione degli accessi, la versione HTTPS e TLS
- Implementa un'applicazione basata su gRPC su un cluster Amazon EKS e accedi ad essa con un Application Load Balancer
- Implementa controlli di accesso preventivi basati sugli attributi per le sottoreti pubbliche
- Distribuisci risorse in una AWS Wavelength zona utilizzando Terraform
- Implementa le automazioni di sicurezza per la AWS WAF soluzione utilizzando Terraform
- Configura un router cellulare serverless per un'architettura basata su celle
- Usa Amazon Q Developer come assistente di programmazione per aumentare la tua produttività
- Visualizza i log e i parametri di AWS Network Firewall utilizzando Splunk

Database e archiviazione

Argomenti

- Database
- Archiviazione e backup

Database

Argomenti

- Accedi alle tabelle Microsoft SQL Server locali da Microsoft SQL Server su Amazon EC2 utilizzando server collegati
- Aggiungi HA a Oracle PeopleSoft su Amazon RDS Custom utilizzando una replica di lettura
- Analizza le dipendenze degli oggetti per le migrazioni parziali del database da Oracle a PostgreSQL
- Valuta le prestazioni delle query per la migrazione dei database SQL Server su MongoDB Atlas su AWS
- Automatizza le blue/green implementazioni dei database globali di Amazon Aurora utilizzando i principi IaC
- Automatizza la replica delle istanze Amazon RDS su Account AWS
- Automatizza le attività del database nell'edizione SQL Server Express in esecuzione su Amazon EC2 utilizzando Task AWS Lambda Scheduler
- Automatizza il failover e il failback tra regioni utilizzando DR Orchestrator Framework
- Esegui automaticamente il backup dei database SAP HANA utilizzando Systems Manager e EventBridge
- Genera automaticamente un modello PynamoDB e funzioni CRUD per Amazon DynamoDB utilizzando un'applicazione Python
- Blocca l'accesso pubblico ad Amazon RDS utilizzando Cloud Custodian
- Configurare l'autenticazione Windows per Amazon RDS for Microsoft SQL Server utilizzando AWS Managed Microsoft AD
- Configurazione dell'accesso multi-account in Amazon DynamoDB
- Configurare il routing di sola lettura in un gruppo di disponibilità Always On in SQL Server su AWS
- Connect utilizzando un tunnel SSH in pGAdmin
- Convertire le query JSON Oracle in SQL del database PostgreSQL
- Copia le tabelle Amazon DynamoDB tra gli account utilizzando AWS Backup
- Copia le tabelle Amazon DynamoDB tra gli account utilizzando un'implementazione personalizzata
- Crea report dettagliati su costi e utilizzo per Amazon RDS e Amazon Aurora
- Distribuisci istanze di cluster di failover di SQL Server su Amazon e EC2 Amazon FSx utilizzando Terraform

- Emula gli array PL/SQL associativi Oracle in Amazon Aurora PostgreSQL e Amazon RDS per PostgreSQL
- Abilita connessioni crittografate per le istanze DB PostgreSQL in Amazon RDS
- Crittografa un'istanza database Amazon RDS for PostgreSQL esistente
- Applica il tagging automatico dei database Amazon RDS al momento del lancio
- Stima del costo di una tabella DynamoDB per la capacità su richiesta
- Stima dei costi di storage per una tabella Amazon DynamoDB
- Stima le dimensioni del motore Amazon RDS per un database Oracle utilizzando i report AWR
- Esporta tabelle Amazon RDS for SQL Server in un bucket S3 utilizzando AWS DMS
- Gestisci blocchi anonimi nelle istruzioni SQL dinamiche in Aurora PostgreSQL
- Aiutaci a far rispettare il tagging di DynamoDB
- Implementa il disaster recovery tra regioni con AWS DMS e Amazon Aurora
- Esegui la migrazione di funzioni e procedure Oracle con più di 100 argomenti a PostgreSQL
- Esegui la migrazione dei carichi di lavoro Redis su Redis Enterprise Cloud su AWS
- Esegui la migrazione da SAP HANA ad AWS utilizzando SAP HSR con lo stesso nome host
- Esegui la migrazione del gruppo di disponibilità Microsoft SQL Server Always On utilizzando AWS Application Migration Service
- Esegui la migrazione di SQL Server su AWS utilizzando gruppi di disponibilità distribuiti
- Migrare un database relazionale su MongoDB Atlas su AWS
- Migrare un ambiente MongoDB ospitato autonomamente su MongoDB Atlas su AWS
- Esegui la migrazione di un database Oracle ad Amazon DynamoDB utilizzando AWS DMS
- Esegui la migrazione da Oracle 8i o 9i ad Amazon RDS for Oracle utilizzando AWS DMS SharePlex
- Esegui la migrazione di un database MySQL locale su Amazon EC2
- Monitora Amazon Aurora per le istanze senza crittografia
- Monitora GoldenGate i log di Oracle utilizzando Amazon CloudWatch
- Ripiattaforma Oracle Database Enterprise Edition alla Standard Edition 2 su Amazon RDS per Oracle
- Replica i database mainframe su AWS utilizzando Precisly Connect
- Pianifica i lavori per Amazon RDS for PostgreSQL e Aurora PostgreSQL utilizzando Lambda e Secrets Manager

- Invia notifiche per un'istanza di database Amazon RDS for SQL Server utilizzando un server SMTP locale e Database Mail
- Configura il disaster recovery per SAP su IBM Db2 su AWS
- Configura una CI/CD pipeline per la migrazione del database utilizzando Terraform
- Configura un cluster di failover di Microsoft SQL Server su Amazon EC2 utilizzando FSx per Windows File Server
- Configurare un' HA/DR architettura per Oracle E-Business Suite su Amazon RDS Custom con un database di standby attivo
- Trasmetti dati da IBM Db2, SAP, Sybase e altri database a MongoDB Atlas su AWS
- Ruoli di transizione per un' PeopleSoft applicazione Oracle su Amazon RDS Custom for Oracle
- Scarica i dati da un cluster Amazon Redshift tra più account su Amazon S3
- Modelli di migrazione del database per carico di lavoro
- Altri modelli

Accedi alle tabelle Microsoft SQL Server locali da Microsoft SQL Server su Amazon EC2 utilizzando server collegati

Tirumala Dasari e Eduardo Valentim, Amazon Web Services

Riepilogo

Questo modello descrive come accedere alle tabelle di database Microsoft SQL Server locali in esecuzione su Microsoft Windows, dai database Microsoft SQL Server in esecuzione o ospitati su istanze Amazon Elastic Compute Cloud EC2 (Amazon) Windows o Linux utilizzando server collegati.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Amazon EC2 con Microsoft SQL Server in esecuzione su AMI Amazon Linux (Amazon Machine Image)
- AWS Direct Connect tra il server Microsoft SQL Server (Windows) locale e l'istanza Windows o Linux EC2

Versioni del prodotto

- SQL Server 2016 o versioni successive

Architecture

Stack tecnologico di origine

- Database Microsoft SQL Server locale in esecuzione su Windows
- Amazon EC2 con Microsoft SQL Server in esecuzione su AMI Windows o AMI Linux

Stack tecnologico Target

- Amazon EC2 con Microsoft SQL Server in esecuzione su AMI Amazon Linux
- Amazon EC2 con Microsoft SQL Server in esecuzione su AMI Windows

Architettura del database di origine e destinazione

Tools (Strumenti)

- Microsoft SQL Server Management Studio (SSMS) è un ambiente integrato per la gestione di un'infrastruttura SQL Server. Fornisce un'interfaccia utente e un gruppo di strumenti con editor di script avanzati che interagiscono con SQL Server.

Epiche

Cambia la modalità di autenticazione in Windows per SQL Server in Windows SQL Server

Operazione	Description	Competenze richieste
Connect a Windows SQL Server tramite SSMS.		DBA
Modificare la modalità di autenticazione in Windows in SQL Server dal menu contestuale (fare clic con il pulsante destro del mouse) per l'istanza di Windows SQL Server.		DBA

Riavviare il servizio Windows MSSQL

Operazione	Description	Competenze richieste
Riavviare il servizio SQL.	1. In SSMS Object Explorer, scegli l'istanza di SQL Server. 2. Apri il menu contestuale (clic con il pulsante destro del mouse).	DBA

Operazione	Description	Competenze richieste
	3. Scegli Riavvia.	

Crea un nuovo accesso e scegli i database a cui accedere in Windows SQL Server

Operazione	Description	Competenze richieste
Nella scheda Sicurezza, apri il menu contestuale (fai clic con il pulsante destro del mouse) per Accesso e seleziona un nuovo accesso.		DBA
Nella scheda Generale, scegli l'autenticazione di SQL Server, inserisci un nome utente, inserisci la password, quindi conferma la password e deseleziona l'opzione per la modifica della password al prossimo accesso.		DBA
Nella scheda Ruoli del server, scegli Pubblico.		DBA
Nella scheda Mappatura utente, scegli il database e lo schema a cui desideri accedere, quindi evidenzia il database per selezionare i ruoli del database.	Seleziona public e db_datare ader per accedere ai dati dalle tabelle del database.	DBA
Scegli OK per creare un utente.		DBA

Aggiungi l'IP di Windows SQL Server al file host di Linux SQL Server

Operazione	Description	Competenze richieste
Connect alla casella Linux SQL Server tramite la finestra del terminale.		DBA
Aprire il file /etc/hosts e aggiungere l'indirizzo IP del computer Windows con SQL Server.		DBA
Salva il file hosts.		DBA

Crea un server collegato su Linux SQL Server

Operazione	Description	Competenze richieste
Creare un server collegato utilizzando le stored procedure master.sys.sp_addlinkedserver e master.dbo.sp_addlinkedsrvlogin.	Per ulteriori informazioni sull'utilizzo di queste stored procedure, vedere la sezione Informazioni aggiuntive.	DBA, Sviluppatore

Verifica il server e i database collegati creati in SSMS

Operazione	Description	Competenze richieste
In Linux SQL Server in SSMS, vai a Linked Servers e aggiorna.		DBA
Espandi i server e i cataloghi collegati creati nel riquadro a sinistra.	Vedrai i database SQL Server selezionati con tabelle e viste.	DBA

Verifica di poter accedere alle tabelle del database di Windows SQL Server

Operazione	Description	Competenze richieste
Nella finestra di interroga zione SSMS, esegui la query: «select top 3* from [sqllin] .dms_sample_win.db o.mlb_data».	Nota che la clausola FROM utilizza una sintassi in quattro parti: computer.database. schema.table (ad esempio, SELECT name "database s» FROM [sqllin] .master.s ys.databases). SQL2 Nel nostro esempio, abbiamo creato un alias per SQL2 nel file hosts, quindi non è necessario inserire il nome NetBIOS effettivo tra parentesi quadre. Se utilizzi i nomi NetBIOS effettivi, tieni presente che AWS utilizza per impostazione predefinita nomi NetBIOS come Win-XXXX e SQL Server richiede parentesi quadre per i nomi con trattini.	DBA, Sviluppatore

Risorse correlate

- Note di rilascio per SQL Server su Linux

Informazioni aggiuntive

Utilizzo di procedure memorizzate per creare server collegati

SSMS non supporta la creazione di server collegati per Linux SQL Server, quindi è necessario utilizzare queste procedure memorizzate per crearli:

```
EXEC master.sys.sp_addlinkedserver @server= N'SQLLIN' , @srvproduct= N'SQL Server'
EXEC master.dbo.sp_addlinkedsrvlogin
 @rmtsrvname=N'SQLLIN',@useself=N'False',@locallogin=NULL,@rmtuser=N'username',@rmtpassword='Te
```

Nota 1: inserisci le credenziali di accesso che hai creato in precedenza in Windows SQL Server nella stored procedure. `master.dbo.sp_addlinkedsrvlogin`

Nota 2: `@server` il nome `SQLLIN` e il nome di immissione del file host `172.12.12.4 SQLLIN` devono essere gli stessi.

È possibile utilizzare questo processo per creare server collegati per i seguenti scenari:

- Da Linux SQL Server a Windows SQL Server tramite un server collegato (come specificato in questo modello)
- Da Windows SQL Server a Linux SQL Server tramite un server collegato
- Da Linux SQL Server a un altro Linux SQL Server tramite un server collegato

Aggiungi HA a Oracle PeopleSoft su Amazon RDS Custom utilizzando una replica di lettura

sampath kathirvel, Amazon Web Services

Riepilogo

Per eseguire la soluzione Oracle PeopleSoft Enterprise Resource Planning (ERP) su Amazon Web Services (AWS), puoi utilizzare Amazon Relational Database Service (Amazon RDS) o Amazon RDS Custom per Oracle, che supporta applicazioni legacy, personalizzate e in pacchetti che richiedono l'accesso al sistema operativo e all'ambiente di database sottostanti. Per i fattori chiave da considerare quando si pianifica una migrazione, consulta le strategie di migrazione del database Oracle in AWS Prescriptive Guidance.

Al momento della stesura di questo documento, RDS Custom for Oracle non supporta l'opzione Multi-AZ, disponibile per Amazon RDS for Oracle come soluzione HA che utilizza la replica dello storage. Questo modello consente invece di ottenere l'HA utilizzando un database in standby che crea e mantiene una copia fisica del database primario. Il modello si concentra sui passaggi per eseguire un database di PeopleSoft applicazioni su Amazon RDS Custom with HA utilizzando Oracle Data Guard per configurare una replica di lettura.

Questo modello modifica anche la replica di lettura in modalità di sola lettura. Avere la replica di lettura in modalità di sola lettura offre ulteriori vantaggi:

- Scaricamento dei carichi di lavoro di sola lettura dal database principale
- Attivazione della riparazione automatica dei blocchi danneggiati recuperando i blocchi integri dal database di standby utilizzando la funzione Oracle Active Data Guard
- Utilizzo della funzionalità Far Sync per mantenere sincronizzato il database di standby remoto senza il sovraccarico prestazionale associato alla trasmissione dei redo log a lunga distanza.

L'utilizzo di una replica in modalità di sola lettura richiede l'opzione Oracle Active Data Guard, che comporta un costo aggiuntivo in quanto è una funzionalità con licenza separata di Oracle Database Enterprise Edition.

Prerequisiti e limitazioni

Prerequisiti

- Un' PeopleSoft applicazione esistente su Amazon RDS Custom. Se non disponi di un'applicazione, consulta lo schema Migrate Oracle PeopleSoft to Amazon RDS Custom.
- Un unico livello di PeopleSoft applicazione. Tuttavia, è possibile adattare questo modello per lavorare con più livelli di applicazione.
- Amazon RDS Custom configurato con almeno 8 GB di spazio di swap.
- Una licenza di database Oracle Active Data Guard per convertire la replica di lettura in modalità di sola lettura e utilizzarla per scaricare le attività di reporting in standby. Per ulteriori informazioni, consulta il listino prezzi commerciale di Oracle Technology.

Limitazioni

- Limitazioni generali e configurazioni non supportate per RDS Custom for Oracle
- Limitazioni associate alle repliche di lettura di Amazon RDS Custom for Oracle

Versioni del prodotto

- Per le versioni del database Oracle supportate da Amazon RDS Custom, consulta RDS Custom for Oracle.
- Per le classi di istanze di Oracle Database supportate da Amazon RDS Custom, consulta Supporto delle classi di istanze DB per RDS Custom for Oracle.

Architecture

Stack tecnologico Target

- Amazon RDS Custom per Oracle
- AWS Secrets Manager
- Oracle Active Data Guard
- Applicazione Oracle PeopleSoft

Architettura Target

Il diagramma seguente mostra un'istanza DB personalizzata di Amazon RDS e una replica di lettura Amazon RDS Custom. La replica di lettura utilizza Oracle Active Data Guard per la replica in un'altra

zona di disponibilità. È inoltre possibile utilizzare la replica di lettura per scaricare il traffico di lettura sul database principale e per scopi di reporting.

Per un'architettura rappresentativa che utilizza Oracle PeopleSoft su AWS, consulta Configurare un' PeopleSoft architettura ad alta disponibilità su AWS.

Tools (Strumenti)

Servizi AWS

- Amazon RDS Custom for Oracle è un servizio di database gestito per applicazioni legacy, personalizzate e confezionate che richiedono l'accesso al sistema operativo e all'ambiente di database sottostanti.
- AWS Secrets Manager ti aiuta a sostituire le credenziali codificate nel codice, comprese le password, con una chiamata API a Secrets Manager per recuperare il segreto a livello di codice. In questo modello, si recuperano le password degli utenti del database da Secrets Manager per `RDS_DATAGUARD` con il nome segreto. `do-not-delete-rds-custom-+<<RDS Resource ID>>+-dg`

Altri strumenti

- Oracle Data Guard ti aiuta a creare, mantenere, gestire e monitorare i database in standby.

Best practice

Per raggiungere l'obiettivo di zero perdite di dati (RPO=0), utilizza la modalità di protezione `MaxAvailability` Data Guard, con l'impostazione redo transport `SYNC+NOAFFIRM` per prestazioni migliori. Per ulteriori informazioni sulla selezione della modalità di protezione del database, consulta la sezione Informazioni aggiuntive.

Epiche

Crea la replica di lettura

Operazione	Description	Competenze richieste
Crea la replica di lettura.	Per creare una replica di lettura dell'istanza DB	DBA

Operazione	Description	Competenze richieste
	personalizzata di Amazon RDS, segui le istruzioni nella documentazione di Amazon RDS e usa l'istanza DB personalizzata di Amazon RDS che hai creato (consulta la sezione Prerequisiti) come database di origine. Per impostazione predefinita, la replica di lettura personalizzata di Amazon RDS viene creata come standby fisico e si trova nello stato montato. Ciò è intenzionale per garantire la conformità con la licenza Oracle Active Data Guard. Questo modello include il codice per la configurazione di un database contenitore multitenant (CDB) o un'istanza non CDB.	

Cambia la modalità di protezione di Oracle Data Guard in MaxAvailability

Operazione	Description	Competenze richieste
Accedi alla configurazione del broker Data Guard sul database principale.	In questo esempio, la replica di lettura personalizzata di Amazon RDS è `RDS_CUSTOM_ORCL_D` per l'istanza non CDB e `RDS_CUSTOM_RDSCDB_B` per l'istanza CDB. I database non CDB	DBA

Operazione	Description	Competenze richieste
	sono `orcl_a` (primari) e (in standby). `orcl_d` I nomi dei database per CDB sono `rdscdb_a` (primario) e `rdscdb_b` (standby). È possibile connettersi alla replica di lettura personalizzata RDS direttamente o tramite il database principale. È possibile trovare il nome del servizio di rete per il database nel `tnsnames.ora` file che si trova nella `$ORACLE_HOME/network/admin` directory. RDS Custom for Oracle inserisce automaticamente queste voci per il database principale e le repliche di lettura. La password dell'`RDS_DATAGUARD` utente è archiviata in AWS Secrets Manager, con nome segreto`do-not-delete-rds-custom-+<<RDS Resource ID>>+-dg`. Per ulteriori informazioni su come connettersi a un'istanza RDS Custom utilizzando la chiave SSH (Secure Shell) recuperata da Secrets Manager, vedere Connessione all'istanza DB	

Operazione	Description	Competenze richieste

	personalizzata RDS tramite SSH.
	
	Per accedere alla configurazione del broker Oracle Data Guard tramite la riga di comando Data Guard (`dgmgrl`), utilizzare il codice seguente.
	
	Non CDB
	
	`$ dgmgrl RDS_DATAGUARD@RDS_CUSTOM_ORCL_D`
	`DGMGRL for Linux: Release 19.0.0.0.0 - Production on Fri Sep 30 22:44:49 2022`
	`Version 19.10.0.0.0`
	`Copyright (c) 1982, 2019, Oracle and/or its affiliates. All rights reserved.`
	`Welcome to DGMGRL, type "help" for information.`
	`Password:`
	`Connected to "ORCL_D"`
	`Connected as SYSDG.`
	`DGMGRL>`
	`DGMGRL> show database orcl_d`
	`Database - orcl_d`
	`Role: PHYSICAL STANDBY`
	`Intended State: APPLY-ON`
	`Transport Lag: 0 seconds (computed 0 seconds ago)`	

Operazione	Description	Competenze richieste

```
Apply Lag: 0 seconds
 (computed 0 seconds
 ago)
Average Apply Rate:
 11.00 KByte/s
Instance(s):
ORCL
SUCCESS
DGMGRL>
```

CDB

```
-bash-4.2$ dgmgrl
 C##RDS_DATAGUARD@R
DS_CUSTOM_RDSCDB_B
DGMGRL for Linux:
 Release 19.0.0.0.0 -
 Production on Wed Jan
 11 20:24:11 2023
Version 19.16.0.0.0
Copyright (c) 1982,
 2019, Oracle and/or its
 affiliates. All rights
 reserved.
Welcome to DGMGRL, type
 "help" for informati
on.
Password:
Connected to "RDSCDB_B
"
Connected as SYSDG.
DGMGRL>
DGMGRL> show database
 rdscdb_b
Database - rdscdb_b
  Role:
 PHYSICAL STANDBY
  Intended State:
 APPLY-ON
```

Operazione	Description	Competenze richieste
	Transport Lag: 0 seconds (computed 1 second ago) Apply Lag: 0 seconds (computed 1 second ago) Average Apply Rate: 2.00 KByte/s Real Time Query: OFF Instance(s): RDSCDB Database Status: SUCCESS DGMGRL>	

Operazione	Description	Competenze richieste

Modifica l'impostazione del trasporto dei log connettendoti a DGMGRL dal nodo primario.	Cambia la modalità di trasporto dei log in`FastSync`, corrispondente all'impostazione di redo transport . `SYNC+NOAFFIRM` Per assicurarti di avere impostazioni valide dopo il cambio di ruolo, modificale sia per il database primario che per il database di standby.
	
	Non CDB
	
	`DGMGRL>`
	`DGMGRL> edit database orcl_d set property logxptmode=fastsync;`
	`Property "logxptmode" updated`
	`DGMGRL> show database orcl_d LogXptMode;`
	`LogXptMode = 'fastsync'`
	`DGMGRL> edit database orcl_a set property logxptmode=fastsync;`
	`Property "logxptmode" updated`
	`DGMGRL> show database orcl_a logxptmode;`
	`LogXptMode = 'fastsync'`
	`DGMGRL>`
	
	CDB
	
	`DGMGRL> edit database rdscdb_b set property`
		DBA

Operazione	Description	Competenze richieste
	logxptmode=fastsync;DGMGRL> edit database rdscdb_b set property logxptmode=fastsync; Property "logxptmode" updated DGMGRL> show database rdscdb_b LogXptMode; LogXptMode = 'fastsync' DGMGRL> edit database rdscdb_a set property logxptmode=fastsync; Property "logxptmode" updated DGMGRL> show database rdscdb_a logxptmode; LogXptMode = 'fastsync' DGMGRL>	

Operazione	Description	Competenze richieste

Cambia la modalità di protezione in. MaxAvailability	Cambia la modalità di protezione in `MaxAvailability` collegandoti a DGMGRL dal nodo principale.
	
	Non CDB
	
	`DGMGRL> edit configuration set protection mode as maxavailability;`
	`Succeeded.`
	`DGMGRL> show configuration;`
	`Configuration - rds_dg`
	`Protection Mode: MaxAvailability`
	`Members:`
	`orcl_a - Primary database`
	`orcl_d - Physical standby database`
	`Fast-Start Failover: Disabled`
	`Configuration Status:`
	`SUCCESS (status updated 38 seconds ago)`
	`DGMGRL>`
	
	CDB
	
	`DGMGRL> show configuration`
	`Configuration - rds_dg`
	`Protection Mode: MaxAvailability`
	`Members:`
	`rdscdb_a - Primary database`
		DBA

Operazione	Description	Competenze richieste
	rdscdb_b - Physical standby database Fast-Start Failover: Disabled Configuration Status: SUCCESS (status updated 57 seconds ago) DGMGRL>	

Modifica lo stato della replica da mount a read-only e abilita Redo Apply

Operazione	Description	Competenze richieste

Stop Redo Apply per il database in standby.	Per impostazione predefinita, la replica di lettura viene creata in `MOUNT` modalità. Per aprirla in modalità di sola lettura, è innanzitutto necessario disattivare Redo Apply collegandosi a `DGMGRL` dal nodo primario o di standby.
	Non CDB
	`DGMGRL> show database orcl_dDGMGRL> show database orcl_d`
	`Database - orcl_d`
	`Role: PHYSICAL STANDBY`
	`Intended State: APPLY-ON`
	`Transport Lag: 0 seconds (computed 1 second ago)`
	`Apply Lag: 0 seconds (computed 1 second ago)`
	`Average Apply Rate: 11.00 KByte/s`
		DBA

Operazione	Description	Competenze richieste
	Real Time Query: OFF Instance(s): ORCL Database Status: SUCCESS DGMGRL> edit database orcl_d set state=app ly-off; Succeeded. DGMGRL> show database orcl_d Database - orcl_d Role: PHYSICAL STANDBY Intended State: APPLY- OFF Transport Lag: 0 seconds (computed 1 second ago) Apply Lag: 42 seconds (computed 1 second ago) Average Apply Rate: (unknown) Real Time Query: OFF Instance(s): ORCL Database Status: SUCCESS DGMGRL> CDB DGMGRL> show configura tionDGMGRL> show configuration Configuration - rds_dg Protection Mode: MaxAvailability Members: rdscdb_a - Primary database	

Operazione	Description	Competenze richieste
	rdscdb_b - Physical standby database Fast-Start Failover: Disabled Configuration Status: SUCCESS (status updated 57 seconds ago) DGMGRL> show database rdscdb_b; Database - rdscdb_b Role: PHYSICAL STANDBY Intended State: APPLY-ON Transport Lag: 0 seconds (computed 1 second ago) Apply Lag: 0 seconds (computed 1 second ago) Average Apply Rate: 2.00 KByte/s Real Time Query: OFF Instance(s): RDSCDB Database Status: SUCCESS DGMGRL> edit database rdscdb_b set state=apply-off; Succeeded. DGMGRL> show database rdscdb_b; Database - rdscdb_b Role: PHYSICAL STANDBY Intended State: APPLY-OFF Transport Lag: 0 seconds (computed 1 second ago)	

Operazione	Description	Competenze richieste
	Apply Lag: 0 seconds (computed 1 second ago) Average Apply Rate: (unknown) Real Time Query: OFF Instance(s): RDSCDB Database Status: SUCCESS	

Operazione	Description	Competenze richieste
Aprire l'istanza di lettura della replica in modalità di sola lettura.	Connettiti al database di standby utilizzando la voce TNS e aprilo in modalità di sola lettura collegandoti al database dal nodo primario o di standby. Non CDB `$ sqlplus RDS_DATAG UARD@RDS_CUSTOM_OR CL_D as sysdg -bash-4.2$ sqlplus RDS_DATAGUARD@RDS_ CUSTOM_ORCL_D as sysdg SQL*Plus: Release 19.0.0.0.0 - Productio n on Fri Sep 30 23:00:14 2022 Version 19.10.0.0.0 Copyright (c) 1982, 2020, Oracle. All rights reserved. Enter password: Last Successful login time: Fri Sep 30 2022 22:48:27 +00:00 Connected to: Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production Version 19.10.0.0.0 SQL> select open_mode from v$database; OPEN_MODE -------------------- MOUNTED SQL> alter database open read only;`	DBA

Operazione	Description	Competenze richieste
	Database altered. SQL> select open_mode from v$database; OPEN_MODE ---------- READ ONLY SQL> CDB -bash-4.2$ sqlplus C##RDS_DATAGUARD@R DS_CUSTOM_RDSCDB_B as sysdg SQL*Plus: Release 19.0.0.0.0 - Productio n on Wed Jan 11 21:14:07 2023 Version 19.16.0.0.0 Copyright (c) 1982, 2022, Oracle. All rights reserved. Enter password: Last Successful login time: Wed Jan 11 2023 21:12:05 +00:00 Connected to: Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production Version 19.16.0.0.0 SQL> select name,open _mode from v$database; NAME OPEN_MODE --------- --------- ---------- RDSCDB MOUNTED SQL> alter database open read only; Database altered.	

Operazione	Description	Competenze richieste
	`SQL> select name,open _mode from v$database; NAME OPEN_MODE --------- ---------- ----------- RDSCDB READ ONLY SQL>`	

Operazione	Description	Competenze richieste
Attiva redo apply sull'istanza di replica letta.	Attiva redo apply sull'istanza di replica letta utilizzando DGMGR L dal nodo primario o di standby. Non CDB `$ dgmgrl RDS_DATAG UARD@RDS_CUSTOM_OR CL_D DGMGRL for Linux: Release 19.0.0.0.0 - Production on Fri Sep 30 23:02:16 2022 Version 19.10.0.0.0 Copyright (c) 1982, 2019, Oracle and/or its affiliates. All rights reserved. Welcome to DGMGRL, type "help" for informati on. Password: Connected to "ORCL_D" Connected as SYSDG. DGMGRL> edit database orcl_d set state=apply-on; DGMGRL> edit database orcl_d set state=app ly-on; Succeeded. DGMGRL> show database orcl_d Database - orcl_d Role: PHYSICAL STANDBY Intended State: APPLY-ON`	DBA

Operazione	Description	Competenze richieste

```
Transport Lag: 0
 seconds (computed 0
 seconds ago)
Apply Lag: 0 seconds
 (computed 0 seconds
 ago)
Average Apply Rate:
 496.00 KByte/s
Real Time Query: ON
Instance(s):
ORCL
Database Status:
SUCCESS
DGMGRL>
```

CDB

```
-bash-4.2$ dgmgrl
 C##RDS_DATAGUARD@R
DS_CUSTOM_RDSCDB_B
-bash-4.2$ dgmgrl
 C##RDS_DATAGUARD@R
DS_CUSTOM_RDSCDB_B
DGMGRL for Linux:
 Release 19.0.0.0.0 -
 Production on Wed Jan
 11 21:21:11 2023
Version 19.16.0.0.0
Copyright (c) 1982,
 2019, Oracle and/or
 its affiliates.  All
 rights reserved.
Welcome to DGMGRL, type
 "help" for informati
on.
Password:
Connected to "RDSCDB_B
"
Connected as SYSDG.
```

Operazione	Description	Competenze richieste
	DGMGRL> edit database rdscdb_b set state=apply-on; Succeeded. DGMGRL> show database rdscdb_b Database - rdscdb_b Role: PHYSICAL STANDBY Intended State: APPLY-ON Transport Lag: 0 seconds (computed 0 seconds ago) Apply Lag: 0 seconds (computed 0 seconds ago) Average Apply Rate: 35.00 KByte/s Real Time Query: ON Instance(s): RDSCDB Database Status: SUCCESS DGMGRL> show database rdscdb_b Database - rdscdb_b Role: PHYSICAL STANDBY Intended State: APPLY-ON Transport Lag: 0 seconds (computed 1 second ago) Apply Lag: 0 seconds (computed 1 second ago) Average Apply Rate: 16.00 KByte/s Real Time Query: ON Instance(s): RDSCDB	

Operazione	Description	Competenze richieste
	`Database Status: SUCCESS DGMGRL>`	

Risorse correlate

- Configurazione di Amazon RDS come PeopleSoft database Oracle (white paper AWS)
- Guida Oracle Data Guard Broker (documentazione di riferimento Oracle)
- Concetti e amministrazione di Data Guard (documentazione di riferimento Oracle)

Informazioni aggiuntive

Seleziona la modalità di protezione del database

Oracle Data Guard offre tre modalità di protezione per configurare l'ambiente Data Guard in base ai requisiti di disponibilità, protezione e prestazioni. La tabella seguente riassume queste tre modalità.

Modalità di protezione	Redo (impostazione di trasporto)	Description
PRESTAZIONI MASSIME	`ASYNC`	Per le transazioni che avvengono sul database primario, i redo data vengono trasmessi e scritti in modo asincrono nel redo log del database di standby. Pertanto, l'impatto sulle prestazioni è minimo. `MaxPerformance` non è possibile fornire RPO=0 a causa della spedizione asincrona dei log.

PROTEZIONE MASSIMA	`SYNC+AFFIRM`	Per le transazioni sul database primario, i redo data vengono trasmessi e scritti in modo sincrono sul redo log on del database di standby prima che la transazione venga confermata. Se il database in standby non è più disponibile, il database primario si chiude automaticamente per garantire la protezione delle transazioni.
DISPONIBILITÀ MASSIMA	`SYNC+AFFIRM`	È simile alla `MaxProtection` modalità, tranne quando non viene ricevuta alcuna conferma dal database di standby. In tal caso, funziona come se fosse in `MaxPerformance` modalità tale da preservare la disponibilità del database primario fino a quando non sarà nuovamente in grado di scrivere il redo stream su un database di standby sincronizzato.

`SYNC+NOAFFIRM`	Per le transazioni sul database primario, il ripristino viene trasmesso in modo sincrono al database di standby e il primario attende solo la conferma che il ripristino è stato ricevuto in standby e non che è stato scritto sul disco di standby. Questa modalità, nota anche come`FastSync`, può offrire un vantaggio in termini di prestazioni a scapito della potenziale esposizione alla perdita di dati in un caso speciale di più errori simultane i.

Le repliche di lettura in RDS Custom for Oracle vengono create con la modalità di protezione delle massime prestazioni, che è anche la modalità di protezione predefinita per Oracle Data Guard. La modalità a prestazioni massime offre il minor impatto sulle prestazioni sul database primario, il che può aiutarti a soddisfare il requisito RPO (Recovery Point Objective) misurato in secondi.

Per raggiungere l'obiettivo di zero perdite di dati (RPO=0), è possibile personalizzare la modalità di protezione di Oracle Data Guard `MaxAvailability` con l'`SYNC+NOAFFIRM`impostazione Redo Transport per migliorare le prestazioni. Poiché i commit sul database primario vengono riconosciuti solo dopo che i vettori di ripristino corrispondenti sono stati trasmessi correttamente al database di standby, la latenza di rete tra l'istanza principale e la replica può essere fondamentale per i carichi di lavoro sensibili al commit. Si consiglia di eseguire test di carico per il carico di lavoro per valutare l'impatto sulle prestazioni quando la replica di lettura è personalizzata per l'esecuzione in modalità. `MaxAvailability`

L'implementazione della replica di lettura nella stessa zona di disponibilità del database principale offre una latenza di rete inferiore rispetto alla distribuzione della replica di lettura in una zona di disponibilità diversa. Tuttavia, l'implementazione della replica primaria e di lettura nella stessa zona di disponibilità potrebbe non soddisfare i requisiti di disponibilità elevata perché, nell'improbabile caso di

indisponibilità della zona di disponibilità, ne risentono sia l'istanza principale che l'istanza di replica di lettura.

Analizza le dipendenze degli oggetti per le migrazioni parziali del database da Oracle a PostgreSQL

Nome utente, Amazon Web Services

Riepilogo

Questo modello descrive l'importanza di identificare e gestire sistematicamente le dipendenze del sistema durante la migrazione di un database Oracle parziale su Amazon Relational Database Service (Amazon RDS) o Amazon Aurora PostgreSQL. In una migrazione parziale, viene migrato solo un sottoinsieme di oggetti e dati del database originale, mentre il database di origine continua a funzionare e servire applicazioni che dipendono da componenti non migrati

È necessario identificare e analizzare l'ambito della migrazione quando si gestiscono database su larga scala con applicazioni strettamente collegate con dipendenze upstream e downstream. Per iniziare una migrazione parziale, identificate gli oggetti dell'ambito, tra cui tabelle, trigger, viste, stored procedure, funzioni e pacchetti. Il processo di identificazione dell'ambito segue un approccio globale:

- Gli oggetti dell'ambito di primo livello vengono identificati tramite riferimenti diretti nel codice dell'applicazione e nei lavori critici specifici del modulo.
- Gli oggetti di secondo livello sono derivati attraverso un'analisi completa delle dipendenze.

Una volta compreso come interagiscono le diverse parti del sistema, è possibile pianificare meglio l'ordine corretto per lo spostamento dei componenti del database e ridurre il rischio di errori di migrazione. La tabella seguente elenca diversi tipi di analisi delle dipendenze.

Tipo di analisi	Aree di interesse	Scopo
Dipendenze dagli oggetti	• Tabelle • Visualizzazioni • Procedure archiviate • Funzioni • Triggers	Identifica le relazioni tra gli oggetti del database e le relative strutture gerarchiche
Dipendenze dei segmenti	• Relazioni con chiave esterna	Mappa le relazioni tra i dati e mantiene l'integrità referenziale

	• Portachiavi primari • Riferimenti tra schemi	
Dipendenze di sicurezza	• Autorizzazioni degli utenti • Gerarchie di ruoli • Privilegi degli oggetti	Garantisce il corretto controllo degli accessi, la migrazion e e la manutenzione della sicurezza
Modelli di accesso	• Operazioni di lettura • Operazioni di scrittura	Determina i modelli di interazio ne del database

Per mantenere la coerenza tra i sistemi di origine e di destinazione, stabilisci meccanismi di sincronizzazione dei dati durante il periodo di transizione. È inoltre necessario modificare il codice e le funzioni dell'applicazione per gestire la distribuzione dei dati tra i database Oracle di origine e PostgreSQL di destinazione.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un database Oracle (origine)
- Un Amazon RDS o Amazon Aurora PostgreSQL (destinazione)

Versioni del prodotto

- Oracle 19c o versioni successive
- PostgreSQL 16 o versione successiva

Architecture

Stack tecnologico di origine

- Oracle 19c o versione successiva

Stack tecnologico Target

- Amazon RDS o Amazon Aurora PostgreSQL

Architettura Target

Il diagramma seguente mostra il processo di migrazione da un database Oracle locale ad Amazon RDS for Oracle, che prevede quanto segue:

- Identificazione delle dipendenze del database
- Migrazione del codice e degli oggetti del database utilizzando AWS Schema Conversion Tool ()AWS SCT
- Migrazione dei dati utilizzando AWS Database Migration Service ()AWS DMS
- Replica delle modifiche in corso tramite Change Data Capture (CDC) utilizzando AWS DMS

Per ulteriori informazioni, consulta Integrazione AWS Database Migration Service con AWS Schema Conversion Tool nella documentazione. AWS

Tools (Strumenti)

Servizi AWS

- Amazon Relational Database Service (Amazon RDS) per Oracle ti aiuta a configurare, gestire e scalare un database relazionale Oracle in. Cloud AWS
- Amazon Aurora è un motore di database relazionale completamente gestito creato per il cloud e compatibile con MySQL e PostgreSQL.
- AWS Schema Conversion Tool (AWS SCT) supporta migrazioni di database eterogenee convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione.
- AWS Database Migration Service (AWS DMS) consente di migrare gli archivi di dati all'interno Cloud AWS o tra combinazioni di configurazioni cloud e locali.

Altri servizi

- Oracle SQL Developer è un ambiente di sviluppo integrato che semplifica lo sviluppo e la gestione dei database Oracle nelle implementazioni tradizionali e basate sul cloud. Per questo modello, è possibile utilizzare SQL*Plus.

Best practice

Per le best practice relative al provisioning e alla migrazione di un database Oracle, consulta Best practice per la migrazione ad Amazon RDS for Oracle.

Epiche

Identifica le dipendenze degli oggetti

Operazione	Description	Competenze richieste
Crea una tabella di oggetti.	Identifica gli oggetti essenziali per la funzionalità dell'applicazione e crea una tabella denominata`DEPENDENT_ANALYSIS_BASELINE`. Aggiungi record per ogni oggetto alla tabella. Per un esempio, consulta la sezione Informazioni aggiuntive.	Ingegnere dei dati, DBA
Crea una procedura di database.	Create una procedura memorizzata denominata `sp_object_dependency_analysis` per analizzare le dipendenze degli oggetti in entrambe le direzioni (avanti e indietro) utilizzando i dati della `DBA_DEPENDENCIES` tabella. Per un esempio, consultate la sezione Informazioni aggiuntive.	Ingegnere dei dati, DBA
Esegui la procedura.	Esegui gli script a ogni livello successivo fino a quando non vengono trovate nuove dipendenze tra gli oggetti. Tutte le dipendenze e i	DBA, ingegnere dei dati

Operazione	Description	Competenze richieste
	livelli vengono memorizza ti nella tabella. `DEPENDENT _ANALYSIS_BASELINE`	

Crea una procedura per le dipendenze a livello di segmento

Operazione	Description	Competenze richieste
Crea una tabella delle dipendenze.	Crea una tabella delle dipendenze a livello di segmento denominata. `REFERENTIAL_ANALYS IS_BASELINE` Quando vengono scoperte tutte le dipendenze a livello di oggetto, controlla le tabelle principali interrogando la tabella. `DEPENDENT _ANALYSIS_BASELINE` `DBA_CONSTRAINT` Escludi le dipendenze in cui le tabelle di base sono referenzi ate da altre tabelle. Il backfilli ng gestisce queste relazioni. Quello che segue è uno script di esempio: `CREATE TABLE REFERENTI AL_ANALYSIS_BASELINE (CHILD_OWNER VARCHAR2( 50 BYTE), CHILD_NAME VARCHAR2( 100 BYTE), PARENT_OWNER VARCHAR2( 50 BYTE),`	Ingegnere dei dati, DBA

Operazione	Description	Competenze richieste
	`PARENT_NAME VARCHAR2( 50 BYTE), REFERENCE_PATH VARCHAR2(1000 BYTE));`	
Crea una procedura di database.	Crea una procedura chiamata `SP_OBJECT_REFERENT IAL_ANALYSIS` e genera un'analisi referenziale per tutti gli oggetti identificati. Per un esempio, consultate la sezione Informazioni aggiuntive.	Ingegnere dei dati, DBA
Esegui la procedura.	Esegui la procedura per ottenere dipendenze referenzi ali. Genera i dettagli dell'ogge tto di analisi referenziale in. `REFERENTIAL_ANALYS IS_BASELINE`	Ingegnere dei dati, DBA

Identifica oggetti che leggono e scrivono

Operazione	Description	Competenze richieste
Crea tabelle per leggere e scrivere oggetti.	Utilizzate il seguente script per creare una tabella di oggetti di lettura denominat a `TABLE_READ_OBJECT_ DETAILS` e una tabella di oggetti di scrittura denominat a`TABLE_WRITE_OBJECT _DETAILS` : `CREATE TABLE TABLE_REA D_OBJECT_DETAILS`	Ingegnere dei dati, DBA

Operazione	Description	Competenze richieste
	(OWNER VARCHAR2(50 BYTE), TAB_NAME VARCHAR2(50 BYTE), READER_OWNER VARCHAR2(50 BYTE), READER_NAME VARCHAR2(50 BYTE), READER_TYPE VARCHAR2(50 BYTE)); CREATE TABLE TABLE_WRITE_OBJECT_DETAILS (TABLE_NAME VARCHAR2(100 BYTE), WRITEOBJ_OWNER VARCHAR2(100 BYTE), WRITEOBJ_NAME VARCHAR2(100 BYTE), WRITEOBJ_TYPE VARCHAR2(100 BYTE), LINE VARCHAR2(100 BYTE), TEXT VARCHAR2(4000 BYTE), OWNER VARCHAR2(50 BYTE));	

Operazione	Description	Competenze richieste
Crea una procedura per l'analisi.	Create rispettivamente le procedure `SP_READER_OBJECTS_ANALYSIS` e `SP_WRITER_OBJECTS_ANALYSIS` quelle per l'analisi degli oggetti di lettura e degli oggetti di scrittura. Queste procedure utilizzano il pattern matching per trovare oggetti correlati. Per un esempio, consultate la sezione Informazioni aggiuntive.	Ingegnere dei dati, DBA
Esegui le procedure.	Eseguite queste procedure per identificare gli oggetti dipendenti.	DBA, ingegnere dei dati

Rivedi i privilegi del database

Operazione	Description	Competenze richieste
Crea una tabella per la revisione dei privilegi.	Crea una tabella per analizzare i privilegi denominati. `OBJECT_PRIVS_ANALYSIS` Per acquisire in modo ricorsivo i privilegi degli oggetti nella `DEPENDENT_ANALYSIS_BASELINE` tabella, utilizzate lo script seguente: `CREATE TABLE OBJECT_PRIVS_ANALYSIS`	Ingegnere dei dati, DBA

Operazione	Description	Competenze richieste
	`(OWNER VARCHAR2(50 BYTE), OBJECT_NAME VARCHAR2(50 BYTE), USER_NAME VARCHAR2(50 BYTE), PRIVS VARCHAR2(50 BYTE));`	
Creare una procedura per la revisione dei privilegi.	Crea una procedura denominata`SP_OBJECT_PRIVS_ANALYSIS` . Genera un'analisi dei privilegi per gli oggetti identificati. Per un esempio, consulta la sezione Informazioni aggiuntive.	DBA, ingegnere dei dati
Esegui la procedura.	Esegui la procedura per acquisirli nella `OBJECT_PRIVS_ANALYSIS` tabella.	DBA, ingegnere dei dati

risoluzione dei problemi

Problema	Soluzione
Impossibile accedere alle tabelle dei dizionari	Assicuratevi che l'utente che ha creato gli oggetti di analisi possa accedere alle tabelle DBA.

Risorse correlate

Documentazione AWS

- Documentazione di Amazon RDS e Aurora

- Playbook di migrazione da Oracle Database 19c ad Amazon Aurora PostgreSQL
- AWS Database Migration Service Che cos'è?
- Cos'è AWS Schema Conversion Tool?

Altra documentazione

- Oggetti del database Oracle

Informazioni aggiuntive

Script per **DEPENDENT_ANALYSIS_BASELINE**

```
CREATE TABLE DEPENDENT_ANALYSIS_BASELINE
(OWNER VARCHAR2(128 BYTE) NOT NULL ENABLE,
OBJECT_NAME VARCHAR2(128 BYTE) NOT NULL ENABLE,
OBJECT_TYPE VARCHAR2(20 BYTE),
DEPEDNCY_LEVEL NUMBER,
PROJECT_NEED VARCHAR2(20 BYTE),
CATAGORY VARCHAR2(4000 BYTE),
COMMENTS VARCHAR2(4000 BYTE),
CATAGORY1 CLOB,
COMMENTS1 CLOB,
CUSTOMER_COMMENTS VARCHAR2(1000 BYTE),
BACKFILL_TO_GUS VARCHAR2(1000 BYTE),
BACKFILL_NEAR_REAL_TIME_OR_BATCH VARCHAR2(1000 BYTE),
PK_EXISTS VARCHAR2(3 BYTE),
UI_EXISTS VARCHAR2(3 BYTE),
LOB_EXISTS VARCHAR2(3 BYTE),
MASTER_LINK VARCHAR2(100 BYTE),
CONSTRAINT PK_DEPENDENT_ANALYSIS_BASELINE PRIMARY KEY (OWNER,OBJECT_NAME,OBJECT_TYPE));
```

Procedura per **SP_WRITER_OBJECTS_ANALYSIS**

```
CREATE OR REPLACE PROCEDURE SP_WRITER_OBJECTS_ANALYSIS IS
BEGIN
  EXECUTE IMMEDIATE 'TRUNCATE TABLE TABLE_WRITE_OBJECT_DETAILS';
  FOR I IN (SELECT OWNER, OBJECT_NAME FROM DEPENDENT_ANALYSIS_BASELINE WHERE
 OBJECT_TYPE = 'TABLE')
  LOOP
    INSERT INTO TABLE_WRITE_OBJECT_DETAILS(OWNER, TABLE_NAME, WRITEOBJ_OWNER,
 WRITEOBJ_NAME, WRITEOBJ_TYPE, LINE, TEXT)
```

```
    SELECT DISTINCT I.OWNER, I.OBJECT_NAME, OWNER WRITEOBJ_OWNER, NAME, TYPE, LINE,
 TRIM(TEXT)
    FROM DBA_SOURCE
    WHERE UPPER(TEXT) LIKE '%' || I.OBJECT_NAME || '%'
      AND (UPPER(TEXT) LIKE '%INSERT%' || I.OBJECT_NAME || '%'
        OR UPPER(TEXT) LIKE '%UPDATE%' || I.OBJECT_NAME || '%'
        OR UPPER(TEXT) LIKE '%DELETE%' || I.OBJECT_NAME || '%'
        OR UPPER(TEXT) LIKE '%UPSERT%' || I.OBJECT_NAME || '%'
        OR UPPER(TEXT) LIKE '%MERGE%' || I.OBJECT_NAME || '%')
      AND UPPER(TEXT) NOT LIKE '%PROCEDURE%'
      AND UPPER(TEXT) NOT LIKE 'PROCEDURE%'
      AND UPPER(TEXT) NOT LIKE '%FUNCTION%'
      AND UPPER(TEXT) NOT LIKE 'FUNCTION%'
      AND UPPER(TEXT) NOT LIKE '%TRIGGER%'
      AND UPPER(TEXT) NOT LIKE 'TRIGGER%'
      AND UPPER(TRIM(TEXT)) NOT LIKE '%AFTER UPDATE%'
      AND UPPER(TRIM(TEXT)) NOT LIKE 'BEFORE UPDATE%'
      AND UPPER(TRIM(TEXT)) NOT LIKE 'BEFORE INSERT%'
      AND UPPER(TRIM(TEXT)) NOT LIKE 'AFTER INSERT%'
      AND UPPER(TRIM(TEXT)) NOT LIKE 'BEFORE DELETE%'
      AND UPPER(TRIM(TEXT)) NOT LIKE 'AFTER DELETE%'
      AND UPPER(TRIM(TEXT)) NOT LIKE '%GGLOGADM.GG_LOG_ERROR%'
      AND (TRIM(TEXT) NOT LIKE '/*%' AND TRIM(TEXT) NOT LIKE '--%' )
      AND (OWNER, NAME, TYPE) IN (SELECT OWNER, NAME, TYPE FROM DBA_DEPENDENCIES WHERE
 REFERENCED_NAME = I.OBJECT_NAME);
  END LOOP;
END;
```

Script per **SP_READER_OBJECTS_ANALYSIS**

```
CREATE OR REPLACE PROCEDURE SP_READER_OBJECTS_ANALYSIS IS
BEGIN
  EXECUTE IMMEDIATE 'TRUNCATE TABLE TABLE_READ_OBJECT_DETAILS';
  FOR I IN (SELECT OWNER, OBJECT_NAME FROM DEPENDENT_ANALYSIS_BASELINE WHERE
 OBJECT_TYPE = 'TABLE')
  LOOP
    INSERT INTO TABLE_READ_OBJECT_DETAILS
    SELECT DISTINCT i.owner, i.object_name, owner, name, type
    FROM dba_dependencies
    WHERE referenced_name = I.OBJECT_NAME
    AND referenced_type = 'TABLE'
    AND type NOT IN ('SYNONYM', 'MATERIALIZED VIEW', 'VIEW')
    AND (owner, name, type) NOT IN (
```

```
      SELECT DISTINCT owner, trigger_name, 'TRIGGER'
      FROM dba_triggers
      WHERE table_name = I.OBJECT_NAME
      AND table_owner = i.owner
      UNION ALL
      SELECT DISTINCT owner, name, type
      FROM dba_source
      WHERE upper(text) LIKE '%' || I.OBJECT_NAME || '%'
      AND (upper(text) LIKE '%INSERT %' || I.OBJECT_NAME || '%'
        OR upper(text) LIKE '%UPDATE% ' || I.OBJECT_NAME || '%'
        OR upper(text) LIKE '%DELETE %' || I.OBJECT_NAME || '%'
        OR upper(text) LIKE '%UPSERT %' || I.OBJECT_NAME || '%'
        OR upper(text) LIKE '%MERGE %' || I.OBJECT_NAME || '%')
      AND upper(text) NOT LIKE '%PROCEDURE %'
      AND upper(text) NOT LIKE 'PROCEDURE %'
      AND upper(text) NOT LIKE '%FUNCTION %'
      AND upper(text) NOT LIKE 'FUNCTION %'
      AND upper(text) NOT LIKE '%TRIGGER %'
      AND upper(text) NOT LIKE 'TRIGGER %'
      AND upper(trim(text)) NOT LIKE 'BEFORE INSERT %'
      AND upper(trim(text)) NOT LIKE 'BEFORE UPDATE %'
      AND upper(trim(text)) NOT LIKE 'BEFORE DELETE %'
      AND upper(trim(text)) NOT LIKE 'AFTER INSERT %'
      AND upper(trim(text)) NOT LIKE 'AFTER UPDATE %'
      AND upper(trim(text)) NOT LIKE 'AFTER DELETE %'
      AND (trim(text) NOT LIKE '/*%' AND trim(text) NOT LIKE '--%'));
  END LOOP;
END;
```

Script per **SP_OBJECT_REFERENTIAL_ANALYSIS**

```
CREATE OR REPLACE PROCEDURE SP_OBJECT_REFERENTIAL_ANALYSIS IS
BEGIN
  EXECUTE IMMEDIATE 'TRUNCATE TABLE REFERENTIAL_ANALYSIS_BASELINE';
  INSERT INTO REFERENTIAL_ANALYSIS_BASELINE
  WITH rel AS (
    SELECT DISTINCT c.owner, c.table_name, c.r_owner r_owner,
      (SELECT table_name FROM dba_constraints
       WHERE constraint_name = c.r_constraint_name
       AND owner = c.r_owner) r_table_name
    FROM dba_constraints c
    WHERE constraint_type = 'R'
    AND c.owner NOT IN (SELECT username FROM dba_users WHERE oracle_maintained = 'Y')
```

```
    AND c.r_owner NOT IN (SELECT username FROM dba_users WHERE oracle_maintained =
 'Y')),
  tab_list AS (
    SELECT OWNER, object_name
    FROM DEPENDENT_ANALYSIS_BASELINE
    WHERE UPPER(OBJECT_TYPE) = 'TABLE')
  SELECT DISTINCT owner child_owner, table_name child, r_owner parent_owner,
    r_table_name parent, SYS_CONNECT_BY_PATH(r_table_name, ' -> ') || ' -> ' ||
 table_name PATH
  FROM rel
  START WITH (r_owner, r_table_name) IN (SELECT * FROM tab_list)
  CONNECT BY NOCYCLE (r_owner, r_table_name) = ((PRIOR owner, PRIOR table_name))
  UNION
  SELECT DISTINCT owner child_owner, table_name child, r_owner parent_owner,
    r_table_name parent, SYS_CONNECT_BY_PATH(table_name, ' -> ') || ' -> ' ||
 r_table_name PATH
  FROM rel
  START WITH (owner, table_name) IN (SELECT * FROM tab_list)
  CONNECT BY NOCYCLE (owner, table_name) = ((PRIOR r_owner, PRIOR r_table_name));
END;
```

Script per **SP_OBJECT_PRIVS_ANALYSIS**

```
CREATE OR REPLACE PROCEDURE SP_OBJECT_PRIVS_ANALYSIS IS
  V_SQL VARCHAR2(4000);
  V_CNT NUMBER;
BEGIN
  V_SQL := 'TRUNCATE TABLE OBJECT_PRIVS_ANALYSIS';
  EXECUTE IMMEDIATE V_SQL;
  FOR I IN (SELECT OWNER, OBJECT_NAME FROM DEPENDENT_ANALYSIS_BASELINE WHERE
 OBJECT_TYPE = 'TABLE')
  LOOP
    INSERT INTO OBJECT_PRIVS_ANALYSIS(OWNER, OBJECT_NAME, USER_NAME, PRIVS)
    WITH obj_to_role AS (
      SELECT DISTINCT GRANTEE role_name,
        DECODE(privilege, 'SELECT', 'READ', 'REFERENCE', 'READ', 'INSERT', 'WRITE',
               'UPDATE', 'WRITE', 'DELETE', 'WRITE', privilege) privs
      FROM DBA_TAB_PRIVS t, DBA_ROLES r
      WHERE OWNER = I.OWNER
      AND TYPE = 'TABLE'
      AND TABLE_NAME = I.OBJECT_NAME
      AND t.GRANTEE = r.ROLE
      AND r.ROLE IN (SELECT ROLE FROM DBA_ROLES WHERE ORACLE_MAINTAINED = 'N')
```

```
    )
    SELECT I.OWNER, I.OBJECT_NAME, grantee, privs
    FROM (
      -- Recursively Role to User mapping with privilege
      SELECT DISTINCT grantee, privs
      FROM (SELECT rp.granted_role, rp.grantee, privs,
        (SELECT DECODE(COUNT(*), 0, 'ROLE', 'USER')
         FROM (SELECT 'User' FROM DBA_users WHERE username = rp.GRANTEE)) grantee_type
        FROM DBA_role_privs rp, obj_to_role r
        WHERE rp.granted_role = r.role_name
        AND grantee IN ((SELECT USERNAME FROM DBA_USERS WHERE ORACLE_MAINTAINED = 'N')
                        UNION (SELECT ROLE FROM DBA_ROLES WHERE ORACLE_MAINTAINED =
 'N'))
        AND granted_role IN (SELECT ROLE FROM DBA_ROLES WHERE ORACLE_MAINTAINED = 'N')
        START WITH granted_role IN (SELECT DISTINCT role_name FROM obj_to_role)
        CONNECT BY granted_role = PRIOR grantee)
      WHERE grantee_type = 'USER'
    )
    UNION
    (
      -- Direct Object grants to User
      SELECT I.OWNER, I.OBJECT_NAME, GRANTEE,
        DECODE(privilege, 'SELECT', 'READ', 'REFERENCE', 'READ', 'INSERT', 'WRITE',
               'UPDATE', 'WRITE', 'DELETE', 'WRITE', privilege) privs
      FROM DBA_TAB_PRIVS, DBA_USERS
      WHERE GRANTEE = USERNAME
      AND OWNER = I.OWNER
      AND TYPE = 'TABLE'
      AND TABLE_NAME = I.OBJECT_NAME
    )
    ORDER BY 2 DESC;
  END LOOP;
END;
```

Procedura per **SP_OBJECT_DEPENDENCY_ANALYSIS**

```
CREATE OR REPLACE PROCEDURE SP_OBJECT_DEPENDENCY_ANALYSIS (v_level NUMBER) IS
  TYPE typ IS RECORD (
    schema VARCHAR2(100),
    obj_type VARCHAR2(100),
    obj_name VARCHAR2(100),
    path VARCHAR2(5000)
  );
```

```
  TYPE array IS TABLE OF typ;
  l_data array;
  c SYS_REFCURSOR;
  l_errors NUMBER;
  l_errno NUMBER;
  l_msg VARCHAR2(4000);
  l_idx NUMBER;
  l_level NUMBER;
BEGIN
  l_level := v_level + 1;
  OPEN c FOR
    WITH obj_list AS (
      SELECT owner schema_name, object_type, object_name
      FROM DEPENDENT_ANALYSIS_BASELINE
      WHERE depedncy_level = v_level
    ),
    fw_dep_objects AS (
      SELECT level lvl, owner, name, type, referenced_owner, referenced_name,
        referenced_type, SYS_CONNECT_BY_PATH(name, ' -> ') || ' -> ' || referenced_name
 PATH
      FROM dba_dependencies
      START WITH (owner, CASE WHEN type = 'PACKAGE BODY' THEN 'PACKAGE' ELSE type END,
 name)
        IN (SELECT schema_name, object_type, object_name FROM obj_list)
      CONNECT BY NOCYCLE (owner, type, name) =
        ((PRIOR referenced_owner, PRIOR referenced_type, PRIOR referenced_name))
    ),
    bw_dep_objects AS (
      SELECT level lvl, owner, name, type, referenced_owner, referenced_name,
        referenced_type, SYS_CONNECT_BY_PATH(name, ' <- ') || ' <- ' || referenced_name
 PATH
      FROM dba_dependencies
      START WITH (referenced_owner, CASE WHEN referenced_type = 'PACKAGE BODY' THEN
 'PACKAGE'
        ELSE referenced_type END, referenced_name) IN (SELECT schema_name, object_type,
 object_name FROM obj_list)
      CONNECT BY NOCYCLE (referenced_owner, referenced_type, referenced_name) =
        ((PRIOR owner, PRIOR type, PRIOR name))
    )
    SELECT * FROM (
      (SELECT DISTINCT referenced_owner schema, referenced_type obj_type,
        referenced_name obj_name, path FROM fw_dep_objects)
      UNION
      (SELECT DISTINCT owner schema, type obj_type, name obj_name, path
```

```
       FROM bw_dep_objects)
   )
   WHERE schema IN (SELECT username FROM all_users WHERE oracle_maintained = 'N')
   ORDER BY obj_type;

 LOOP
   FETCH c BULK COLLECT INTO l_data LIMIT 100;
   BEGIN
     FORALL i IN 1..l_data.count SAVE EXCEPTIONS
       INSERT INTO DEPENDENT_ANALYSIS_BASELINE (
         owner, object_name, object_type, catagory, depedncy_level, project_need,
comments
       )
       VALUES (
         l_data(i).schema,
         l_data(i).obj_name,
         CASE WHEN l_data(i).obj_type = 'PACKAGE BODY' THEN 'PACKAGE' ELSE
l_data(i).obj_type END,
         'level ' || l_level || ' dependency',
         l_level,
         '',
         'from dependency proc' || l_data(i).path
       );
   EXCEPTION
     WHEN OTHERS THEN
       l_errors := sql%bulk_exceptions.count;
       FOR i IN 1..l_errors LOOP
         l_errno := sql%bulk_exceptions(i).error_code;
         l_msg := SQLERRM(-l_errno);
         l_idx := sql%bulk_exceptions(i).error_index;
         UPDATE DEPENDENT_ANALYSIS_BASELINE
         SET catagory1 = catagory1 || ', found in level' || l_level || ' dependent of
' || l_data(l_idx).path,
             comments1 = comments1 || ', from dependency proc exception ' ||
l_data(i).path
         WHERE owner = l_data(l_idx).schema
         AND object_name = l_data(l_idx).obj_name
         AND object_type = l_data(l_idx).obj_type;
       END LOOP;
   END;
   EXIT WHEN c%NOTFOUND;
 END LOOP;
 CLOSE c;
```

```
END;
```

Valuta le prestazioni delle query per la migrazione dei database SQL Server su MongoDB Atlas su AWS

Battulga Purevragcha, Amazon Web Services

Krishnakumar PeerIslands Sathyanarayana, US Inc

Babu Srinivasan, MongoDB

Riepilogo

Questo modello fornisce indicazioni per caricare MongoDB con dati simili a quelli reali e valutare le prestazioni delle query di MongoDB che si avvicinano il più possibile allo scenario di produzione. La valutazione fornisce input per aiutarti a pianificare la migrazione a MongoDB da un database relazionale. Il modello utilizza PeerIslands Test Data Generator e Performance Analyzer per testare le prestazioni delle query.

Questo modello è particolarmente utile per la migrazione di Microsoft SQL Server a MongoDB, poiché eseguire trasformazioni dello schema e caricare dati dalle istanze correnti di SQL Server a MongoDB può essere molto complesso. Invece, puoi caricare dati quasi reali in MongoDB, comprendere le prestazioni di MongoDB e perfezionare la progettazione dello schema prima di iniziare la migrazione vera e propria.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Familiarità con MongoDB Atlas
- Schema MongoDB di destinazione
- Schemi di interrogazione tipici

Limitazioni

- I tempi di caricamento dei dati e le prestazioni saranno limitati dalla dimensione dell'istanza del cluster MongoDB. Ti consigliamo di scegliere istanze consigliate per l'uso in produzione per comprendere le prestazioni del mondo reale.

- PeerIslands Test Data Generator and Performance Analyzer attualmente supportano solo caricamenti e interrogazioni di dati online. L'elaborazione batch offline (ad esempio, il caricamento di dati in MongoDB utilizzando i connettori Spark) non è ancora supportata.
- PeerIslands Test Data Generator and Performance Analyzer supportano le relazioni tra campi all'interno di una raccolta. Non supporta le relazioni tra le raccolte.

Edizioni del prodotto

- Questo modello supporta sia MongoDB Atlas che MongoDB Enterprise Advanced.

Architecture

Stack tecnologico Target

- MongoDB Atlas o MongoDB Enterprise Advanced

Architettura

PeerIslands Test Data Generator and Performance Analyzer è stato creato utilizzando Java e Angular e archivia i dati generati su Amazon Elastic Block Store (Amazon EBS). Lo strumento è composto da due flussi di lavoro: generazione di dati di test e test delle prestazioni.

- Nella generazione dei dati di test, si crea un modello, che è la rappresentazione JSON del modello di dati che deve essere generato. Dopo aver creato il modello, è possibile generare i dati in una raccolta di destinazione, come definito dalla configurazione di generazione del carico.
- Nei test delle prestazioni, si crea un profilo. Un profilo è uno scenario di test in più fasi in cui è possibile configurare le operazioni di creazione, lettura, aggiornamento ed eliminazione (CRUD), le pipeline di aggregazione, il peso per ogni operazione e la durata di ogni fase. Dopo aver creato il profilo, è possibile eseguire test delle prestazioni sul database di destinazione in base alla configurazione.

PeerIslands Test Data Generator and Performance Analyzer archivia i dati su Amazon EBS, in modo da poter connettere Amazon EBS a MongoDB utilizzando qualsiasi meccanismo di connessione supportato da MongoDB, tra cui peering, elenchi di autorizzazione ed endpoint privati. Per impostazione predefinita, lo strumento non include componenti operativi; tuttavia, può essere

configurato con Amazon Managed Service for Prometheus, Amazon Managed Grafana, Amazon CloudWatch e AWS Secrets Manager, se necessario.

Tools (Strumenti)

- PeerIslands Test Data Generator and Performance Analyzer include due componenti. Il componente Test Data Generator ti aiuta a generare dati reali altamente specifici per il cliente in base al tuo schema MongoDB. Lo strumento è completamente basato sull'interfaccia utente con una ricca libreria di dati e può essere utilizzato per generare rapidamente miliardi di record su MongoDB. Lo strumento fornisce anche funzionalità per implementare relazioni tra i campi nello schema MongoDB. Il componente Performance Analyzer ti aiuta a generare query e aggregazioni altamente specifiche per il cliente ed eseguire test realistici delle prestazioni su MongoDB. Puoi utilizzare Performance Analyzer per testare le prestazioni di MongoDB con profili di carico avanzati e query parametrizzate per il tuo caso d'uso specifico.

Best practice

Consulta le seguenti risorse:

- Best practice per la progettazione di schemi MongoDB (sito Web per sviluppatori MongoDB)
- Best practice per la distribuzione di MongoDB Atlas su AWS (sito Web MongoDB)
- Connessione sicura delle applicazioni a un piano dati MongoDB Atlas con AWS PrivateLink (post sul blog AWS)
- Guida alle migliori pratiche per le prestazioni di MongoDB (sito Web MongoDB)

Epiche

Comprendi i tuoi dati di origine

Operazione	Description	Competenze richieste
Comprendi l'impronta del database dell'attuale sorgente di SQL Server.	Comprendi il tuo attuale footprint di SQL Server. Ciò può essere ottenuto eseguendo query sullo `INFORMATION` schema del database. Determina il numero	DBA

Operazione	Description	Competenze richieste
	di tabelle e le dimensioni di ciascuna tabella. Analizza l'indice associato a ciascuna tabella. Per ulteriori informazioni sull'analisi SQL, consulta il post di blog SQL2Mongo: Data Migration Journey sul PeerIslands sito Web.	
Comprendi lo schema sorgente.	Determina lo schema della tabella e la rappresentazione aziendale dei dati (ad esempio codici postali, nomi e valuta). Utilizza il diagramma ER (Entity Relationship) esistente o genera il diagramma ER dal database esistente. Per ulteriori informazioni, consulta il post del blog SQL2Mongo : Data Migration Journey sul sito web. PeerIslands	DBA
Comprendi i modelli di interrogazione.	Documenta le 10 principali query SQL che utilizzi. Puoi utilizzare le tabelle performance_schema.events_statements_summary_by_digest disponibili nel database per comprendere le query principali. Per ulteriori informazioni, consulta il post del blog Mongo: Data Migration Journey sul SQL2 sito PeerIslands web.	DBA

Operazione	Description	Competenze richieste
Comprendi gli impegni SLA.	Documenta gli accordi sui livelli di servizio (SLAs) previsti per le operazioni del database. Le misure tipiche includono la latenza delle query e le query al secondo. Le misure e i relativi obiettivi sono in genere disponibili nei documenti sui requisiti non funzionali (NFR).	DBA

Definire lo schema MongoDB

Operazione	Description	Competenze richieste
Definire lo schema di destinazione.	Definisci varie opzioni per lo schema MongoDB di destinazione. Per ulteriori informazioni, consulta Schemi nella documentazione di MongoDB Atlas. Considerate le migliori pratiche e i modelli di progettazione basati sulle relazioni tra tabelle.	Ingegnere MongoDB
Definisci modelli di query target.	Definisci le query e le pipeline di aggregazione di MongoDB. Queste query sono l'equivalente delle principali query acquisite per il carico di lavoro di SQL Server. Per capire come costruire pipeline di aggregazione MongoDB, consulta la documentazione di MongoDB.	Ingegnere MongoDB

Operazione	Description	Competenze richieste
Definire il tipo di istanza MongoDB.	Determina la dimensione dell'istanza che intendi utilizzare per il test. Per informazioni, consulta la documentazione di MongoDB.	Ingegnere MongoDB

Preparare il database di destinazione

Operazione	Description	Competenze richieste
Configura il cluster MongoDB Atlas.	Per configurare un cluster MongoDB su AWS, segui le istruzioni nella documentazione di MongoDB.	Ingegnere MongoDB
Crea utenti nel database di destinazione.	Configura il cluster MongoDB Atlas per l'accesso e la sicurezza della rete seguendo le istruzioni nella documentazione di MongoDB.	Ingegnere MongoDB
Crea ruoli appropriati in AWS e configura il controllo degli accessi basato sui ruoli per Atlas.	Se necessario, configura altri utenti seguendo le istruzioni nella documentazione di MongoDB. Configura l'autenticazione e l'autorizzazione tramite i ruoli AWS.	Ingegnere MongoDB
Configura Compass per l'accesso a MongoDB Atlas.	Configura l'utilità MongoDB Compass GUI per facilitare la navigazione e l'accesso.	Ingegnere MongoDB

Imposta il carico di base utilizzando Test Data Generator

Operazione	Description	Competenze richieste
Installa Test Data Generator.	Installa PeerIsland Test Data Generator nel tuo ambiente.	Ingegnere MongoDB
Configura Test Data Generator per generare i dati appropriati.	Crea un modello utilizzando la libreria di dati per generare dati specifici per ogni campo dello schema MongoDB. Per ulteriori informazioni, consulta MongoDB Data Generator & Perf. Video sull'analizzatore.	Ingegnere MongoDB
Generatore di dati di test scalabile orizzontalmente per generare il carico richiesto.	Utilizza il modello che hai creato per avviare la generazione del carico sulla raccolta di destinazione configurando il parallelismo richiesto. Determina i tempi e la scala per generare i dati necessari.	Ingegnere MongoDB
Convalida il carico in MongoDB Atlas.	Controlla i dati caricati in MongoDB Atlas.	Ingegnere MongoDB
Genera gli indici richiesti su MongoDB.	Definisci gli indici come richiesto, in base ai modelli di query. Per le best practice, consulta la documentazione di MongoDB.	Ingegnere MongoDB

Effettuare test delle prestazioni

Operazione	Description	Competenze richieste
Configura i profili di carico in Performance Analyzer.	Crea un profilo di test delle prestazioni in Performance Analyzer configurando query specifiche e il peso, la durata dell'esecuzione del test e le fasi corrispondenti. Per ulteriori informazioni, consulta MongoDB Data Generator & Perf. Video sull'analizzatore.	Ingegnere MongoDB
Esegui test delle prestazioni.	Utilizza il profilo di test delle prestazioni che hai creato per avviare il test rispetto alla raccolta target configurando il parallelismo richiesto. Scala orizzontalmente lo strumento di test delle prestazioni per eseguire query su MongoDB Atlas.	Ingegnere MongoDB
Registra i risultati dei test.	Registra la latenza P95, P99 per le query.	Ingegnere MongoDB
Ottimizza lo schema e i modelli di query.	Modifica gli indici e i modelli di query per risolvere eventuali problemi di prestazioni.	Ingegnere MongoDB

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse AWS temporanee.	Elimina tutte le risorse temporanee utilizzate per Test	Amministratore AWS

Operazione	Description	Competenze richieste
	Data Generator e Performan ce Analyzer.	
Aggiorna i risultati dei test delle prestazioni.	Comprendi le prestazioni delle query di MongoDB e confrontale con le tue. SLAs Se necessario, perfeziona lo schema MongoDB ed esegui nuovamente il processo.	Ingegnere MongoDB
Concludi il progetto.	Chiudi il progetto e fornisci feedback.	Ingegnere MongoDB

Risorse correlate

- GitHub archivio: S3ToAtlas
- Schema: progettazione dello schema MongoDB
- Pipeline di aggregazione: pipeline di aggregazione MongoDB
- Dimensionamento MongoDB Atlas: selezione del livello di dimensionamento
- Video: Generatore di dati MongoDB e Perf. Analizzatore
- Riferimenti: documentazione MongoDB
- Tutorial: guida per sviluppatori MongoDB, MongoDBJumpstart
- AWS Marketplace: MongoDB Atlas su AWS Marketplace
- Soluzioni per partner AWS: MongoDB Atlas sulla distribuzione di riferimento AWS

Risorse aggiuntive:

- Analisi SQL
- Forum della comunità di sviluppatori MongoDB
- Domande sull'ottimizzazione delle prestazioni di MongoDB
- Analisi operativa con Atlas e Redshift
- Modernizzazione delle applicazioni con MongoDB Atlas e AWS Elastic Beanstalk

Automatizza le blue/green implementazioni dei database globali di Amazon Aurora utilizzando i principi IaC

Ishwar Chauthaiwale, ANKIT JAIN e Ramu Jagini, Amazon Web Services

Riepilogo

La gestione degli aggiornamenti, delle migrazioni o degli sforzi di scalabilità dei database può essere difficile per le organizzazioni che eseguono carichi di lavoro critici sui database globali di Amazon Aurora. Garantire che queste operazioni vengano eseguite senza interruzioni, senza interruzioni, è essenziale per mantenere la disponibilità del servizio ed evitare interruzioni per gli utenti.

Una strategia di blue/green implementazione offre una soluzione a questa sfida in quanto consente di eseguire due ambienti identici contemporaneamente: blu (l'ambiente corrente) e verde (il nuovo ambiente). Una blue/green strategia consente di implementare modifiche, eseguire test e spostare il traffico tra ambienti con rischi e tempi di inattività minimi.

Questo modello consente di automatizzare il processo di blue/green distribuzione per i database globali Aurora utilizzando i principi dell'infrastruttura come codice (IaC). Utilizza AWS CloudFormationAmazon Route 53 e Amazon Route 53 per semplificare le blue/green distribuzioni. AWS Lambda Per migliorare l'affidabilità, utilizza gli identificatori di transazione globali (GTIDs) per la replica. La replica basata su GTID offre una migliore coerenza dei dati e funzionalità di failover tra ambienti rispetto alla replica con log binario (binlog).

> **Note**
>
> Questo modello presuppone che si stia utilizzando un cluster di database globale Edition compatibile con Aurora MySQL. Se invece utilizzi Aurora compatibile con PostgreSQL, utilizza gli equivalenti PostgreSQL dei comandi MySQL.

Seguendo i passaggi di questo schema, puoi:

- Esegui il provisioning di un database globale Aurora verde: utilizzando CloudFormation i modelli, crei un ambiente verde che rispecchia l'ambiente blu esistente.
- Imposta la replica basata su GTID: configuri la replica GTID per mantenere sincronizzati gli ambienti blu e verdi.
- Cambia traffico senza interruzioni: utilizzi Route 53 e Lambda per passare automaticamente il traffico dall'ambiente blu a quello verde dopo la sincronizzazione completa.

- Finalizza la distribuzione: verifichi che l'ambiente verde sia pienamente operativo come database principale, quindi interrompi la replica e pulisci eventuali risorse temporanee.

L'approccio in questo modello offre i seguenti vantaggi:

- Riduce i tempi di inattività durante gli aggiornamenti o le migrazioni critiche del database: l'automazione garantisce una transizione fluida tra gli ambienti con interruzioni minime del servizio.
- Consente un ripristino rapido: se si verifica un problema dopo che il traffico è passato all'ambiente verde, è possibile tornare rapidamente all'ambiente blu e mantenere la continuità del servizio.
- Migliora i test e le verifiche: l'ambiente verde può essere completamente testato senza influire sull'ambiente blu in tempo reale, il che riduce la probabilità di errori nella produzione.
- Garantisce la coerenza dei dati: la replica basata su GTID mantiene sincronizzati gli ambienti blu e verde, prevenendo la perdita di dati o le incongruenze durante la migrazione.
- Mantiene la continuità aziendale: l'automazione delle blue/green implementazioni aiuta a evitare lunghe interruzioni e perdite finanziarie mantenendo i servizi disponibili durante gli aggiornamenti o le migrazioni.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Un cluster di database globale di origine compatibile con Aurora MySQL (ambiente blu). I database globali forniscono una configurazione multiregionale per l'elevata disponibilità e il disaster recovery. Per istruzioni sulla configurazione di un cluster di database globale, consulta la documentazione di Aurora.
- Replica basata su GTID abilitata nel cluster di origine.

Limitazioni

- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- Aurora, compatibile con MySQL 8.0 o versione successiva

Architecture

Il diagramma illustra quanto segue:

- Configurazione globale del database: un cluster di database globale Aurora viene distribuito strategicamente su due. Regioni AWS Questa configurazione consente la distribuzione geografica e la ridondanza regionale per funzionalità avanzate di disaster recovery.
- Replica da regione primaria a regione secondaria: il meccanismo di replica logica garantisce una perfetta sincronizzazione dei dati dalla regione primaria alla regione secondaria. Questa replica mantiene la coerenza dei dati con una latenza minima su distanze geografiche.
- Replica basata su GTID tra cluster: la replica basata su GTID mantiene la coerenza transazionale e il flusso di dati ordinato tra il cluster primario blu e il cluster primario verde e garantisce una sincronizzazione affidabile dei dati.
- Replica blu da principale a secondaria: la replica logica stabilisce una solida pipeline di dati tra il cluster primario blu e il relativo cluster secondario. Questa replica consente la sincronizzazione continua dei dati e l'elevata disponibilità.
- Configurazione DNS di Route 53: i record delle zone ospitate di Route 53 gestiscono la risoluzione DNS per tutti gli endpoint blu e verdi del database del cluster. Questa configurazione fornisce una mappatura degli endpoint senza interruzioni e consente un routing efficiente del traffico durante gli scenari di failover.

Tools (Strumenti)

Servizi AWS

- Amazon Aurora è un motore di database relazionale completamente gestito creato per il cloud e compatibile con MySQL e PostgreSQL.
- CloudFormationti aiuta a modellare e configurare AWS le tue risorse in modo da poter dedicare meno tempo alla gestione di tali risorse e più tempo a concentrarti sulle applicazioni in esecuzione. AWS Crei un modello che descrive tutte le AWS risorse che desideri e si CloudFormation occupa del provisioning e della configurazione di tali risorse per te.

- AWS Lambdaè un servizio di elaborazione che supporta l'esecuzione di codice senza fornire o gestire server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo.
- Amazon Route 53 è un servizio Web DNS altamente scalabile e disponibile.

Best practice

Si consiglia di esaminare attentamente la AWS documentazione per approfondire la comprensione della strategia di distribuzione blu/verde, della replica basata su GTID e delle politiche di routing ponderate in Route 53. Questa conoscenza è fondamentale per implementare e gestire efficacemente le migrazioni dei database, garantire la coerenza dei dati e ottimizzare il routing del traffico. Acquisendo una comprensione completa di queste AWS funzionalità e best practice, sarai meglio attrezzato per gestire gli aggiornamenti futuri, ridurre al minimo i tempi di inattività e mantenere un ambiente di database resiliente e sicuro.

Per le linee guida sull'utilizzo di Servizi AWS for this pattern, consulta la seguente documentazione: AWS

- Le migliori pratiche con Amazon Aurora MySQL
- CloudFormation migliori pratiche
- Le migliori pratiche per l'utilizzo delle AWS Lambda funzioni
- Le migliori pratiche per Amazon Route 53

Epiche

Crea l'ambiente verde

Operazione	Description	Competenze richieste
Crea un backup istantaneo dal cluster blu.	In una blue/green distribuzione, l'ambiente verde rappresenta una nuova versione identica dell'ambiente di database corrente (blu). L'ambiente verde viene utilizzato per testare in	DBA

Operazione	Description	Competenze richieste
	sicurezza gli aggiornamenti, convalidare le modifiche e garantire la stabilità prima di passare al traffico di produzion e. Funge da base di partenza per l'implementazione delle modifiche al database con interruzioni minime dell'ambi ente live. Per creare un ambiente verde, devi prima creare un'istantanea del cluster primario (blu) nel database globale compatibile con Aurora MySQL. Questa istantanea funge da base per la creazione dell'ambiente verde. Per creare un'istantanea: 1. Accedi Console di gestione AWS e apri la console Amazon Relational Database Service (Amazon RDS). 2. Seleziona il tuo cluster primario (blu). 3. Scegli Azioni, Scatta un'istantanea. 4. Fornisci un nome per l'istantanea, ad esempio`blue-green-`	

Operazione	Description	Competenze richieste
	demo, e avvia il processo di backup. In alternativa, puoi usare il AWS Command Line Interface (AWS CLI) per creare l'istantanea: `aws rds create-db-cluster-snapshot --db-cluster-snapshot-identifier blue-green-demo --db-cluster-identifier ex-global-cluster --region eu-west-1` Assicurati che l'istantanea sia stata completata correttamente prima di procedere al passaggio successivo.	

Operazione	Description	Competenze richieste
Genera il CloudFormation modello per il tuo database globale e le sue risorse.	Il generatore CloudFormation IAc consente di generare CloudFormation modelli a partire da AWS risorse esistenti. Usa questa funzionalità per creare un CloudFormation modello per il tuo database globale compatibile con Aurora MySQL esistente e le risorse associate. Questo modello configura gruppi di sottoreti, gruppi di sicurezza, gruppi di parametri e altre impostazioni. 1. Segui le istruzioni nella CloudFormation documentazione per accedere allo strumento e collegarlo al tuo AWS ambiente. 2. Seleziona il database globale Aurora e le risorse associate per generare il modello.	DBA

Operazione	Description	Competenze richieste
Modifica il CloudFormation modello per l'ambiente verde.	Personalizza il CloudForm ation modello in modo che rifletta le impostazioni per l'ambiente verde. Ciò include l'aggiornamento dei nomi e degli identificatori delle risorse per garantire che l'ambient e verde funzioni indipende ntemente dal cluster blu. 1. Aggiorna le `DBInstanc eIdentifier` proprietà `DBClusterIdentifie r` and per rappresentare l'ambiente verde. 2. Modifica i nomi di altre risorse (ad esempio, gruppi di sottoreti e gruppi di sicurezza) per evitare conflitti con l'ambiente blu esistente. 3. Abilita la replica basata su GTID nel modello configurando i parametri corretti, come descritto nella documentazione di Aurora. 4. Modifica la `SnapshotI dentifier` proprietà per specificare l' Regione AWS ID del tuo account e il nome dell'istantanea del passaggio precedente:	DBA

Operazione	Description	Competenze richieste
	`SnapshotIdentifier: arn:aws:rds:<region>:<account-id>:snapshot:<snapshot-name>` **Note** Se utilizzate la `SnapshotIdentifier` proprietà per ripristinare un cluster DB, evitate di specificare proprietà come `GlobalClusterIdentifier` `MasterUsername` , o. `MasterUserPassword`	

Operazione	Description	Competenze richieste
Implementa lo CloudFormation stack per creare risorse per l'ambiente verde.	In questo passaggio, si distribuisce il CloudFormation modello personalizzato per creare le risorse per l'ambiente verde. Per distribuire lo stack: CloudFormation 1. Apri la AWS CloudFormation console. 2. In alto a destra, scegli Crea stack, Con nuove risorse (standard). 3. Carica il CloudFormation modello modificato o specifica l'URL del modello. Scegli Next (Successivo). 4. Immettete un nome per lo stack`GreenClusterStack` , ad esempio, e fornite tutti i parametri necessari (ad esempio,`GreenClusterIdentifier`). Scegli Next (Successivo). 5. Configura le opzioni di stack aggiuntive secondo necessità e seleziona la casella per confermare che CloudFormation potrebbe creare AWS Identity and Access Management (IAM)	DBA

Operazione	Description	Competenze richieste
	resources. Scegli Next (Successivo). 6. Esamina i dettagli dello stack. 7. Seleziona Invia. CloudFormation avvia il processo di creazione delle risorse ambientali verdi. Il completamento di questo processo potrebbe richiedere alcuni minuti.	

Operazione	Description	Competenze richieste
Convalida lo CloudFormation stack e le risorse.	Una volta completata l'implementazione CloudFormation dello stack, dovrai verificare che l'ambiente verde sia stato creato correttamente: 1. Nella sezione Output dello CloudFormation stack, controlla gli endpoint del cluster di database e dell'istanza di database per verificare la corretta configurazione. 2. Apri la console Amazon RDS e conferma che il nuovo cluster di database Aurora (ambiente verde) è disponibile. 3. Assicurati che tutte le risorse associate, come sottoreti e gruppi di sicurezza, siano state create e collegate all'ambiente verde. Dopo la verifica, l'ambiente verde è pronto per un'ulteriore configurazione, inclusa la replica dall'ambiente blu.	DBA

Configura la replica basata su GTID

Operazione	Description	Competenze richieste
Verifica le impostazioni GTID sul cluster blu.	GTIDs forniscono un metodo altamente affidabile per replicare i dati tra gli ambienti blu e verdi. La replica basata su GTID offre un approccio resiliente e semplificato assegnando un identificatore univoco a ogni transazione nell'ambiente blu. Questo metodo garantisce che la sincronizzazione dei dati tra gli ambienti sia fluida, coerente e più facile da gestire rispetto alla tradizionale replica binlog. Prima di configurare la replica, è necessario assicurarsi che la replica basata su GTID sia abilitata correttamente sul cluster blu. Questo passaggio garantisce che ogni transazione nell'ambiente blu sia tracciata in modo univoco e possa essere replicata nell'ambiente verde. Per confermare che GTID è abilitato: 1. Sulla console Amazon RDS, esamina il gruppo di parametri assegnato al cluster blu.	DBA

Operazione	Description	Competenze richieste
	2. Verifica che siano impostati i seguenti parametri: • `gtid-mode = ON` • `enforce_gtid_consistency = ON` Queste impostazioni abilitano il tracciamento GTID per tutte le transazioni future nell'ambiente blu. Dopo aver confermato queste impostazioni, puoi iniziare a configurare la replica.	

Operazione	Description	Competenze richieste
Crea un utente di replica.	Per replicare i dati dall'ambiente blu all'ambiente verde, è necessario creare un utente di replica dedicato sul cluster blu. Questo utente sarà responsabile della gestione del processo di replica. Per configurare l'utente di replica: 1. Connect al blue cluster utilizzando un client MySQL. 2. Esegui i seguenti comandi per creare l'utente di replica: `CREATE USER 'repl_user'@'%' IDENTIFIED BY 'repl_password'; GRANT REPLICATION SLAVE ON . TO 'repl_user'@'%'; FLUSH PRIVILEGES;` Questo utente dispone ora delle autorizzazioni necessarie per replicare i dati tra i due ambienti.	DBA

Operazione	Description	Competenze richieste
Configura la replica basata su GTID sul cluster verde.	Il passaggio successivo consiste nel configurare il cluster verde per la replica basata su GTID. Questa configurazione garantisc e che l'ambiente verde rispecchi continuamente tutte le transazioni che avvengono nell'ambiente blu. Per configurare il cluster verde: 1. Connect al cluster verde utilizzando un client MySQL. 2. Esegui il comando seguente per configurare la replica: `CHANGE MASTER TO MASTER_HOST='blue-cluster-endpoint', MASTER_USER='repl_user', MASTER_PASSWORD='repl_password', MASTER_AUTO_POSITION=1;` dove: • Sostituisci `blue-cluster-endpoint` con l'endpoint del tuo cluster blu. • L'`MASTER_AUTO_POSITI`	DBA

Operazione	Description	Competenze richieste
	0N=1 impostazione indica a MySQL di utilizzare la replica basata su GTID. Posiziona automaticamente il cluster verde per replicare le transazio ni del cluster blu senza dover tracciare manualmente i log e le posizioni.	
Avvia la replica sul cluster verde.	È ora possibile avviare il processo di replica. Sul cluster verde, esegui il comando: `START SLAVE;` Ciò consente all'ambiente verde di iniziare a sincroniz zare i dati e a ricevere e applicare transazioni dall'ambi ente blu.	DBA

Operazione	Description	Competenze richieste
Verifica il processo di replica.	Per verificare che l'ambient e verde stia replicando accuratamente i dati dal cluster blu: 1. Esegui il comando seguente sul cluster verde per verificare lo stato della replica: `SHOW SLAVE STATUS\G;` 2. Esamina l'output per verificare quanto segue: • `Slave_IO_Running = Yes` • `Slave_SQL_Running = Yes` • I `Executed_Gtid_Set` valori `Retrieved _Gtid_Set` up-to-date e sono sincronizzati con il cluster blu. • Non ci sono errori di replica sul `Last_Error` campo. Se tutti gli indicatori sono corretti, la replica basata su GTID funziona senza problemi e l'ambiente verde è completamente sincronizzato con l'ambiente blu.	DBA

Passa il traffico dal cluster blu a quello verde

Operazione	Description	Competenze richieste
Configura le politiche di routing ponderate di Route 53.	Dopo aver verificato la coerenza dei dati tra gli ambienti blu e verde, puoi spostare il traffico dal cluster blu al cluster verde. Questa transizione dovrebbe essere fluida e dovrebbe ridurre al minimo i tempi di inattivit à e garantire l'integrità del database dell'applicazione. Per soddisfare questi requisiti , puoi utilizzare Route 53 per il routing DNS e Lambda per automatizzare il cambio di traffico. Inoltre, un piano di rollback ben definito garantisc e la possibilità di tornare al cluster blu in caso di problemi. Il primo passo consiste nel configurare il routing ponderato in Route 53. Il routing ponderato consente di controllare la distribuzione del traffico tra i cluster blu e verdi e di spostare gradualmente il traffico da un ambiente all'altro . Per configurare il routing ponderato: 1. Apri la console Route 53 e scegli la tua zona ospitata.	AWS DevOps

Operazione	Description	Competenze richieste
	2. Crea due record DNS (CNAMEs) per il database: un record per il cluster blu e un record per il cluster verde. 3. Assegna pesi iniziali: • Imposta un peso iniziale basso (ad esempio il 5%) per il cluster verde per inviare una piccola porzione di traffico per i test. • Imposta un peso maggiore (ad esempio il 95 percento) per il cluster blu, in modo che trattenga la maggior parte del traffico. Questa configurazione consente di eseguire una transizione graduale che riduce i rischi e supporta test in tempo reale prima della conversione completa. Per ulteriori informazioni sulle politiche di routing ponderate, consulta la documentazione di Route 53.	

Operazione	Description	Competenze richieste

Implementa una funzione Lambda per monitorare il ritardo di replica.	Per garantire che l'ambient e verde sia completamente sincronizzato con l'ambiente blu, implementa una funzione Lambda che monitora il ritardo di replica tra i cluster. Questa funzione può controllare lo stato della replica, in particola re la metrica Seconds_B ehind_Master, per determina re se il cluster verde è pronto a gestire tutto il traffico.
	
	Ecco un esempio di funzione Lambda che puoi usare:
	
	`import boto3`
	
	`def check_replication_ lag(event, context):`
	`client = boto3.cli ent('rds')`
	`response = client.de scribe_db_instance s(DBInstanceIdenti fier='green-cluster- instance')`
	`replication_status = response['DBInstan ces'][0]['ReadRepl icaDBInstanceIdent ifiers']`
	`if replicati on_status:`
	`lag = replicati on_status[0]['Repl icationLag']`
	`return lag`
		AWS DevOps

Operazione	Description	Competenze richieste
	`return -1` Questa funzione controlla il ritardo di replica e restituisce il valore. Se il ritardo è zero, il cluster verde è completamente sincronizzato con il cluster blu.	

Operazione	Description	Competenze richieste

Automatizza la regolazione del peso del DNS utilizzando Lambda.	Quando il ritardo di replica raggiunge lo zero, è il momento di trasferire tutto il traffico sul cluster verde. Puoi automatizzare questa transizione utilizzando un'altra funzione Lambda che regola i pesi DNS in Route 53 per indirizzare il 100% del traffico verso il cluster verde.
	
	Ecco un esempio di funzione Lambda che automatizza l'interruttore del traffico:
	
	`import boto3`
	
	`def switch_traffic(event, context):`
	`route53 = boto3.client('route53')`
	`lag = check_replication_lag(event, context)`
	`if lag == 0:`
	`response = route53.change_resource_record_sets(`
	`HostedZoneId='YOUR_HOSTED_ZONE_ID',`
	`ChangeBatch={`
	`'Changes': [`
	`{`
	`'Action': 'UPSERT',`
		AWS DevOps

Operazione	Description	Competenze richieste

	'ResourceRecordSet': {
	'Name': 'db.example.com',
	'Type': 'CNAME',
	'SetIdentifier': 'GreenCluster',
	'Weight': 100,
	'TTL': 60,
	'ResourceRecords': [{'Value': 'green-cluster-endpoint'}]
	}
	},
	{
	'Action': 'UPSERT',
	'ResourceRecordSet': {
	'Name': 'db.example.com',
	'Type': 'CNAME',
	'SetIdentifier': 'BlueCluster',
	'Weight': 0,
	'TTL': 60,
	'ResourceRecords':	

Operazione	Description	Competenze richieste

```
 [{'Value': 'blue-clu
ster-endpoint'}]

 }
                    }
                ]
            }
        )
        return response
```

Questa funzione controlla il ritardo di replica e aggiorna i pesi DNS di Route 53 quando il ritardo è zero per spostare completamente il traffico verso il cluster verde.

> **Note**
>
> Durante il processo di cutover, se il cluster blu presenta un traffico di scrittura intenso, prendi in considera zione la possibili tà di sospendere temporaneamente le operazioni di scrittura durante il cutover. Ciò garantisce che la replica recuperi il ritardo e previene le incongruenze dei dati tra i cluster blu e verdi.

Operazione	Description	Competenze richieste
Verifica l'interruttore del traffico.	Dopo che la funzione Lambda ha regolato i pesi DNS, è necessario verificare che tutto il traffico sia indirizzato al cluster verde e che il passaggio sia avvenuto correttamente. Per verificare: 1. Monitora i record DNS di Route 53 per confermare che il traffico venga indirizzato verso il cluster verde. Per ulteriori informazioni, consulta la documentazione di Route 53. 2. Controlla le prestazioni dell'applicazione confermando che gli utenti siano serviti dall'ambiente verde. 3. Verifica le connessioni al database per confermare che il cluster verde stia gestendo tutte le richieste del database. 4. Monitora i CloudWatch parametri di Amazon per rilevare eventuali segni di latenza, ritardo nella replica o peggioramento delle prestazioni. Per ulteriori informazioni, consulta la documentazione di Aurora.	AWS DevOps

Operazione	Description	Competenze richieste
	Se tutto funziona come previsto, l'interruttore del traffico è completo.	

Operazione	Description	Competenze richieste
In caso di problemi, ripristina le modifiche.	Avere un piano di rollback è fondamentale in caso di problemi dopo il cambio di traffico. Ecco come tornare rapidamente al cluster blu, se necessario: 1. Ripristina i pesi DNS in Route 53: utilizza la stessa funzione Lambda o regola manualmente i pesi DNS di Route 53 per reindirizzare il 100% del traffico verso il cluster blu. 2. Monitora le prestazioni delle applicazioni: monitora immediatamente i log delle applicazioni, le CloudWatch metriche e le prestazioni del database per confermare che il passaggio all'ambiente blu abbia risolto i problemi. 3. Identifica e risolvi i problemi: analizza e risolvi eventuali problemi con il cluster verde prima di tentare un altro cambio di traffico. Implementando questo piano di rollback, puoi garantire interruzioni minime agli utenti in caso di problemi imprevisti.	AWS DevOps

Convalida e interrompi la replica basata su GTID

Operazione	Description	Competenze richieste
Interrompi la replica basata su GTID sul cluster verde.	Dopo aver spostato il traffico dall'ambiente blu a quello verde, è necessario convalida re il successo della transizio ne e assicurarsi che il cluster verde funzioni come previsto. Inoltre, la replica basata su GTID tra i cluster blu e verdi deve essere interrott a, poiché l'ambiente verde ora funge da database primario. Il completamento di queste attività garantisc e che l'ambiente sia sicuro, semplificato e ottimizzato per le operazioni in corso. Per interrompere la replica: 1. Usa un client MySQL per connetterti al cluster verde. 2. Esegui il seguente comando SQL per interrompere il processo di replica sul cluster verde: `STOP SLAVE;` 3. (Facoltativo) Se lo si desidera, è possibile ripristinare la configurazione di replica per cancellare eventuali impostazioni di replica residue:	DBA

Operazione	Description	Competenze richieste
	`RESET SLAVE ALL;` Quando si interrompe la replica, il cluster verde diventa completamente indipende nte e funge da ambiente di database principale per i carichi di lavoro.	

Operazione	Description	Competenze richieste
Eliminare le risorse.	La pulizia di tutte le risorse temporanee o inutilizzate create durante la migrazione dal cluster blu a quello verde garantisce che l'ambiente rimanga ottimizzato, sicuro ed economico. La pulizia include la regolazione delle impostazioni di sicurezza, l'esecuzione dei backup finali e la disattivazione delle risorse non necessarie. Per ripulire le risorse: 1. Aggiorna i gruppi di sicurezza: configura i gruppi di sicurezza associati ai cluster blu e verde in modo che riflettano il nuovo ambiente primario (verde). Limita l'accesso all'ambiente blu se non è più necessario e verifica che le impostazioni di sicurezza del cluster verde seguano le migliori pratiche. 2. Effettua un backup finale del cluster verde: una volta completata la migrazione, scatta un'istantanea finale del cluster verde da utilizzare come backup. È possibile utilizzare questa istantanea	AWS DevOps

Operazione	Description	Competenze richieste

	per ripristinare l'ambiente in futuro, se necessario.
	
	`aws rds create-db-snapshot --db-instance-identifier green-cluster-instance --db-snapshot-identifier green-cluster-final-snapshot`
	
	3. Rivedi e rimuovi risorse temporanee: esamina tutte le risorse temporanee create durante la migrazione, ad esempio gruppi di sicurezza temporanei, istantanee o altre configurazioni. Elimina le risorse che non sono più necessarie per evitare costi inutili. Ad esempio, elimina il cluster blu se non è più necessario:
	
	`aws rds delete-db-cluster --db-cluster-identifier blue-cluster-identifier --skip-final-snapshot`
	
	La pulizia delle risorse aiuta a mantenere un ambiente sicuro e semplificato, riduce i costi e garantisce che rimanga solo l'infrastruttura necessaria.	

Risorse correlate

CloudFormation:

- CloudFormation Guida per l'utente
- CloudFormation migliori pratiche
- Genera modelli da risorse esistenti con iAc Generator
- Importa intere applicazioni in CloudFormation (post AWS del blog)

Amazon Aurora:

- Guida per l'utente di Amazon Aurora
- Gestione di un cluster Amazon Aurora DB

Strategia di implementazione blu/verde:

- Panoramica delle implementazioni di Amazon Aurora Blue/Green

Replica basata su GTID:

- Utilizzo della replica basata su GTID (documentazione Amazon RDS)

AWS Lambda:

- AWS Lambda Guida per gli sviluppatori
- Le migliori pratiche per lavorare con le funzioni AWS Lambda

Amazon Route 53:

- Guida per gli sviluppatori di Amazon Route 53
- Routing ponderato

Strumenti client MySQL:

- PyMySQL

Automatizza la replica delle istanze Amazon RDS su Account AWS

Parag Nagwekar e Arun Chandapillai, Amazon Web Services

Riepilogo

Questo modello mostra come automatizzare il processo di replica, tracciamento e ripristino delle istanze DB di Amazon Relational Database Service (Amazon RDS) tra diverse istanze database utilizzando e. Account AWS AWS Step Functions AWS Lambda Puoi utilizzare questa automazione per eseguire repliche su larga scala di istanze DB RDS senza alcun impatto sulle prestazioni o sovraccarico operativo, indipendentemente dalle dimensioni dell'organizzazione. È inoltre possibile utilizzare questo modello per aiutare l'organizzazione a rispettare le strategie obbligatorie di governance dei dati o i requisiti di conformità che richiedono la replica e la ridondanza dei dati su aree diverse. Account AWS Regioni AWS La replica su più account dei dati di Amazon RDS su larga scala è un processo manuale inefficiente e soggetto a errori che può essere costoso e dispendioso in termini di tempo, ma l'automazione in questo modello può aiutarti a ottenere la replica tra account in modo sicuro, efficace ed efficiente.

Prerequisiti e limitazioni

Prerequisiti

- Due Account AWS
- Un'istanza DB RDS, attiva e funzionante nell'origine Account AWS
- Un gruppo di sottoreti per l'istanza DB RDS nella destinazione Account AWS
- Una chiave AWS Key Management Service (AWS KMS) creata nell'origine Account AWS e condivisa con l'account di destinazione (per ulteriori informazioni sui dettagli delle politiche, consulta la sezione Informazioni aggiuntive di questo modello).
- Un AWS KMS key nella destinazione Account AWS per crittografare il database nell'account di destinazione

Limitazioni

- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- Python 3.9 (utilizzo) AWS Lambda
- PostgreSQL 11.3, 13.x e 14.x

Architecture

Stack tecnologico

- Amazon Relational Database Service (Amazon RDS)
- Amazon Simple Notification Service (Amazon SNS)
- AWS Key Management Service (AWS KMS)
- AWS Lambda
- Gestione dei segreti AWS
- AWS Step Functions

Architettura Target

Il diagramma seguente mostra un'architettura per l'utilizzo di Step Functions per orchestrare la replica pianificata e su richiesta delle istanze DB RDS da un account di origine (account A) a un account di destinazione (account B).

Nell'account di origine (account A nel diagramma), la macchina a stati Step Functions esegue le seguenti operazioni:

1. Crea un'istantanea dall'istanza DB RDS nell'account A.
2. Copia e crittografa lo snapshot con un account di origine A. Per garantire la crittografia in transito, lo snapshot viene crittografato indipendentemente AWS KMS key dal fatto che l'istanza DB sia crittografata o meno.
3. Condivide lo snapshot DB con l'account B dando all'account B l'accesso allo snapshot.
4. Invia una notifica all'argomento SNS, quindi l'argomento SNS richiama la funzione Lambda nell'account B.

Nell'account di destinazione (account B nel diagramma), la funzione Lambda esegue la macchina a stati Step Functions per orchestrare quanto segue:

1. Copia l'istantanea condivisa dall'account A all'account B, mentre utilizza prima l'account AWS KMS key dall'account A per decrittografare i dati e poi crittografa i dati utilizzando l'account B. AWS KMS key
2. Legge il segreto da Secrets Manager per acquisire il nome dell'istanza DB corrente.
3. Ripristina l'istanza DB dallo snapshot con un nuovo nome e un valore predefinito per AWS KMS key Amazon RDS.
4. Legge l'endpoint del nuovo database e aggiorna il segreto in Secrets Manager con il nuovo endpoint del database, quindi etichetta l'istanza DB precedente in modo che possa essere eliminata in un secondo momento.
5. Mantiene le ultime N istanze dei database ed elimina tutte le altre istanze.

Tools (Strumenti)

Servizi AWS

- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale in. Cloud AWS
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.
- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su e. Account AWS Regioni AWS
- AWS Key Management Service (AWS KMS) consente di creare e controllare chiavi crittografiche per proteggere i dati.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS SDK per Python (Boto3)è un kit di sviluppo software che ti aiuta a integrare la tua applicazione, libreria o script Python con. Servizi AWS
- Con Gestione dei segreti AWS puoi sostituire le credenziali nel codice,incluse le password, con una chiamata API a Secrets Manager in modo da recuperare il segreto a livello di codice.
- AWS Step Functionsè un servizio di orchestrazione serverless che consente di combinare funzioni Lambda e altro Servizi AWS per creare applicazioni aziendali critiche.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub Crossaccount RDS Replication.

Epiche

Automatizza la replica delle istanze DB RDS con un solo clic Account AWS

Operazione	Description	Competenze richieste
Implementa lo CloudFormation stack nell'account di origine.	1. Accedi all'account Console di gestione AWS di origine (account A) e apri la CloudFormation console. 2. Nel riquadro di navigazione selezionare Stacks (Stack). 3. Scegli Crea stack, quindi scegli Con risorse esistenti (importa risorse). 4. Nella pagina Identifica risorse, scegli Avanti. 5. Nella pagina Specificare il modello, seleziona Carica un modello. 6. Scegli file, seleziona il **`Cloudformation-SourceAccountRDS.yaml`** file dal repository GitHub Crossaccount RDS Replication, quindi scegli Avanti. 7. Per il nome dello stack, inserisci un nome per lo stack. 8. Nella sezione Parametri , specificate i parametri definiti nel modello dello stack:	Amministratore del cloud, architetto del cloud

Operazione	Description	Competenze richieste
	• Per DestinationAccount Number, inserisci il numero di account per l'istanza DB RDS di destinazione. • Per KeyName, inserisci il tuo. AWS KMS key • Per ScheduleExpression , inserisci un'espressione cron (l'impostazione predefinita è ogni giorno alle 12:00). • Per Source DBIdentifier, inserisci il nome del database di origine. • Per DBSnapshotNome sorgente, immettete il nome dell'istantanea o accettate quello predefinito. 9. Scegli Next (Successivo). 10. Nella pagina Configura le opzioni dello stack, lascia i valori predefiniti, quindi scegli Avanti. 11. Controlla la configurazione dello stack, quindi scegli Invia. 12. Scegli la scheda Risorse per il tuo stack, quindi annota l'Amazon Resource Name (ARN) dell'argomento SNS.	

Operazione	Description	Competenze richieste
Implementa lo CloudFormation stack nell'account di destinazione.	1. Accedi all'account Console di gestione AWS di destinazione (account B) e apri la CloudFormation console. 2. Nel riquadro di navigazione selezionare Stacks (Stack). 3. Scegli Crea stack, quindi scegli Con risorse esistenti (importa risorse). 4. Nella pagina Identifica risorse, scegli Avanti. 5. Nella pagina Specificare il modello, seleziona Carica un modello. 6. Scegli il file, seleziona il **`Cloudformation-DestinationAccountRDS.yaml`** file dal repository GitHub Crossaccount RDS Replication, quindi scegli Avanti. 7. Per il nome dello stack, inserisci un nome per lo stack. 8. Nella sezione Parametri, specificate i parametri definiti nel modello dello stack: • Per DatabaseName, inserisci un nome per il tuo database.	Architetto cloud, DevOps ingegnere, amministratore del cloud

Operazione	Description	Competenze richieste
	• Per Engine, inserisci il tipo di motore di database che corrisponde al database di origine. • Per DBInstanceClass, inserisci il tipo di istanza di database preferito o accetta quello predefinito. • Per i sottoreti, inserisci il sottogruppo di sottoreti VPC esistente. Per istruzioni sulla creazione di un gruppo di sottoreti, consulta Fase 2: Creare un gruppo di sottoreti DB nella documentazione di Amazon RDS. • Per SecretName, inserisci il percorso e il nome segreto o accetta quello predefinito. • Per SGID, inserisci l'ID del gruppo di sicurezza del cluster di destinazione. • Per KMSKey, inserisci l'ARN della chiave KMS nel tuo account di destinazione. • Ad esempio NoOfOlder Instances, inserisci il numero di vecchie copie delle istanze DB RDS che	

Operazione	Description	Competenze richieste
	desideri conservare per il rollback. 9. Scegli Next (Successivo). 10. Nella pagina Configura le opzioni dello stack, lascia i valori predefiniti, quindi scegli Avanti. 11. Controlla la configurazione dello stack, quindi scegli Invia. 12. Scegli la scheda Risorse per il tuo stack, quindi annota l'ID fisico e l'`InvokeStepFunction` ARN di.	
Verifica la creazione dell'istanza DB RDS nell'account di destinazione.	1. Accedi Console di gestione AWS e apri la console Amazon RDS. 2. Nel riquadro di navigazione, scegli Database, quindi verifica che la nuova istanza DB RDS compaia nel nuovo cluster.	Amministratore cloud, architetto cloud, DevOps ingegnere

Operazione	Description	Competenze richieste

Sottoscrivi la funzione Lambda all'argomento SNS.	È necessario eseguire i seguenti comandi AWS Command Line Interface (AWS CLI) per sottoscrivere la funzione Lambda nell'account di destinazione (account B) all'argomento SNS nell'account di origine (account A).
	
	Nell'account A, esegui il seguente comando:
	
	`aws sns add-permission \ --label lambda-access --aws-account-id <DestinationAccount> \ --topic-arn <Arn of SNSTopic > \ --action-name Subscribe ListSubscriptionsBy Topic`
	
	Nell'account B, esegui il seguente comando:
	
	`aws lambda add-permission \ --function-name <Name of InvokeStepFunction> \ --source-arn <Arn of SNSTopic > \ --statement-id function-with-sns \ --action lambda:InvokeFunction \`
		Amministratore cloud, architetto cloud, DBA

Operazione	Description	Competenze richieste

	`--principal sns.amazonaws.com`
	
	Nell'account B, esegui il seguente comando:
	
	`aws sns subscribe \ --protocol "lambda" \ --topic-arn <Arn of SNSTopic> \ --notification-endpoint <Arn of InvokeStepFunction>`	

Operazione	Description	Competenze richieste
Sincronizza l'istanza DB RDS dall'account di origine con l'account di destinazione.	Avvia la replica del database su richiesta avviando la macchina a stati Step Functions nell'account di origine. 1. Apri la console Step Functions. 2. Nel riquadro di navigazione, scegli Macchine a stati. 3. Scegli la tua macchina a stati. 4. Nella scheda Esecuzioni, seleziona la funzione, quindi scegli Avvia esecuzione per avviare il flusso di lavoro. **Note** È disponibile uno scheduler che consente di eseguire la replica automaticamente nei tempi previsti, ma per impostazione predefinita è disattivato. Puoi trovare il nome della CloudWatch regola Amazon per lo scheduler nella scheda Risorse dello CloudFormation	Architetto cloud, DevOps ingegnere, amministratore del cloud

Operazione	Description	Competenze richieste
	stack nell'account di destinazione. Per istruzioni su come modificare la regola degli CloudWatch eventi, consulta Eliminazione o disabilitazione di una regola relativa CloudWatch agli eventi nella documentazione. CloudWatch	
Ripristina il database su una delle copie precedenti quando necessario.	1. Apri la console Secrets Manager. 2. Dall'elenco dei segreti, scegli il segreto che hai creato utilizzando il CloudFormation modello precedente. L'applicazione utilizza il segreto per accedere al database nel cluster di destinazione. 3. Per aggiornare il valore segreto dalla pagina dei dettagli, nella sezione Valore segreto, scegli Recupera valore segreto, quindi scegli Modifica. 4. Inserisci i dettagli dell'endpoint del database.	Amministratore cloud, DBA, ingegnere DevOps

Risorse correlate

- Repliche di lettura tra regioni (documentazione Amazon RDS)
- Implementazioni blu/verdi (documentazione Amazon RDS)

Informazioni aggiuntive

Puoi utilizzare la seguente policy di esempio per condividere le tue informazioni. AWS KMS key Account AWS

```
{
    "Version": "2012-10-17",
    "Id": "cross-account-rds-kms-key",
    "Statement": [
        {
            "Sid": "Enable user permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::<SourceAccount>:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
            "Sid": "Allow administration of the key",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::<DestinationAccount>:root"
            },
            "Action": [
                "kms:Create*",
                "kms:Describe*",
                "kms:Enable*",
                "kms:List*",
                "kms:Put*",
                "kms:Update*",
                "kms:Revoke*",
                "kms:Disable*",
                "kms:Get*",
                "kms:Delete*",
                "kms:ScheduleKeyDeletion",
                "kms:CancelKeyDeletion"
```

```
            ],
            "Resource": "*"
        },
        {
            "Sid": "Allow use of the key",
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::<DestinationAccount>:root",
                    "arn:aws:iam::<SourceAccount>:root"
                ]
            },
            "Action": [
                "kms:Encrypt",
                "kms:Decrypt",
                "kms:ReEncrypt*",
                "kms:GenerateDataKey*",
                "kms:DescribeKey",
                "kms:CreateGrant"
            ],
            "Resource": "*"
        }
    ]
}
```

Automatizza le attività del database nell'edizione SQL Server Express in esecuzione su Amazon EC2 utilizzando Task AWS Lambda Scheduler

Subhani Shaik, Amazon Web Services

Riepilogo

Questo modello dimostra come pianificare e gestire le attività del database nell'edizione SQL Server Express, che è la versione gratuita di SQL Server. Tuttavia, l'edizione SQL Server Express non dispone del servizio SQL Server Agent che in genere gestisce le operazioni automatizzate del database. Questo modello spiega come utilizzare Task Scheduler e Lambda come alternativa per automatizzare le attività di database nell'edizione SQL Server Express in esecuzione su un'istanza Amazon Elastic Compute Cloud (EC2Amazon).

Task Scheduler è un'utilità di sistema Windows integrata che facilita l'esecuzione automatica delle attività di routine. Fornisce un meccanismo per pianificare e gestire le operazioni automatizzate, eliminando la necessità di interventi manuali nei processi ricorrenti. AWS Lambdaè un servizio di elaborazione serverless che esegue automaticamente il codice in risposta agli eventi, senza richiedere la gestione dell'infrastruttura sottostante.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un cloud privato virtuale (VPC) creato con Amazon Virtual Private Cloud (Amazon VPC)
- Un' EC2 istanza Amazon con Windows Server
- Volumi Amazon Elastic Block Store (Amazon EBS) collegati a un'istanza EC2 Amazon con Windows Server
- File binari di SQL Server Express Edition

Limitazioni

- Per informazioni sulle limitazioni delle funzionalità dell'edizione SQL Server Express, vedere il sito Web Microsoft.
- Alcune Servizi AWS non sono disponibili in tutte Regioni AWS. Per la disponibilità regionale, vedi AWS Servizi per regione. Per endpoint specifici, consulta Service endpoints and quotas e scegli il link relativo al servizio.

Versioni del prodotto

- SQL Server 2016 o versione successiva con SQL Server Express

Architecture

Il diagramma seguente mostra un' EC2 istanza Amazon in esecuzione con l'edizione SQL Server Express installata. L'istanza è accessibile tramite il client Remote Desktop Protocol (RDP) o da. AWS Systems Manager Session Manager AWS Key Management Service (AWS KMS) gestisce la crittografia dei dati per i volumi Amazon EBS per garantire la data-at-rest sicurezza. L'infrastruttura include anche AWS Identity and Access Management (IAM), che fornisce il controllo degli accessi e gestisce le autorizzazioni per l'esecuzione delle funzioni Lambda. Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) archivia le funzioni Lambda.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Block Store (Amazon EBS) fornisce volumi di storage a livello di blocco da utilizzare con le istanze Amazon. EC2
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Systems Manager Session Managerè uno strumento completamente gestito AWS Systems Manager . Con Session Manager, puoi gestire le tue EC2 istanze Amazon, i dispositivi edge, i server locali e le macchine virtuali ()VMs.

- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.

Altri strumenti

- Microsoft SQL Server Management Studio (SSMS) è uno strumento per la gestione di SQL Server, che include l'accesso, la configurazione e l'amministrazione dei componenti di SQL Server.
- Python è un linguaggio di programmazione per computer generico. È possibile utilizzarlo per creare applicazioni, automatizzare attività e sviluppare servizi su. Cloud AWS
- Task Scheduler è uno strumento Microsoft che puoi utilizzare per pianificare automaticamente le attività di routine sul tuo computer.

Best practice

- Le migliori pratiche per Amazon EC2
- Le migliori pratiche per la distribuzione di Microsoft SQL Server su Amazon EC2
- Le migliori pratiche per lavorare con le AWS Lambda funzioni
- Best practice per la sicurezza in IAM

Epiche

Crea un' EC2 istanza Amazon e installa l'edizione SQL Server Express

Operazione	Description	Competenze richieste
Implementa un' EC2 istanza Amazon.	Per creare un' EC2 istanza Amazon, apri la EC2 console Amazon all'indirizzo https://console.aws.amazon.com/ec2/e seleziona un'Amazon Machine Image (AMI) dall'elenco di istanze disponibili per Windows Server.	DBA, AWS DevOps

Operazione	Description	Competenze richieste
	Per ulteriori informazioni, consulta Launch an Amazon EC2 istance nella AWS documentazione.	

Operazione	Description	Competenze richieste
Installa l'edizione SQL Server Express.	Per installare l'edizione SQL Server Express, completare i seguenti passaggi: 1. Per connetterti alla tua EC2 istanza Amazon, scegli un'opzione: • Opzione A: utilizza Remote Desktop Protocol (RDP). Per istruzioni, consulta Connect alla tua istanza Windows utilizzando un client RDP nella AWS documentazione. • Opzione B: utilizza la EC2 console Amazon e AWS Systems Manager Session Manager. Per istruzioni, consulta Connect alla tua EC2 istanza Amazon utilizzando Session Manager nella AWS documentazione. 2. Per scaricare l'edizione di SQL Server Express richiesta, vai alla sezione Download di SQL Server sul sito Web di Microsoft. 3. Per installare l'edizione SQL Server Express, segui le istruzioni in Pianificare un'installazione di SQL	DBA, AWS DevOps

Operazione	Description	Competenze richieste
	Server sul sito Web di Microsoft.	

Crea attività automatizzate di manutenzione del database

Operazione	Description	Competenze richieste
Identifica le attività di routine.	Identifica le attività di routine che desideri automatizzare. Ad esempio, le seguenti attività sono idonee per l'automazione: • Backup del database (completo, differenziale e registro delle transazioni) • Manutenzione e riorganiz zazione degli indici • Aggiornamenti delle statistic he • Operazioni specifiche dell'applicazione • Pulizia o archiviazione dei dati	DBA
Preparare gli script SQL.	Per preparare gli script SQL, procedi come segue: 1. Crea query SQL per ogni attività di manutenzione. Di seguito è riportato un esempio di query T-SQL per eseguire un backup specifico del database:	DBA

Operazione	Description	Competenze richieste
	`Backup Database <Database_Name> To Disk='C:\Backups\Database_Name.bak'` 2. Salva il file di script come`<File Name>.sql`. Quindi, salva gli script in una posizione accessibile sull'unità locale del server sull' EC2 istanza Amazon o su una condivisione di file di rete.	

Operazione	Description	Competenze richieste
Configura le autorizzazioni di accesso.	Per configurare le autorizza zioni di accesso, procedi come segue: 1. Imposta le autorizza zioni appropriate per il file system. Per istruzion i, vedere Configurare le autorizzazioni del file system per l'accesso al motore di database sul sito Web di Microsoft. 2. Verificare che l'account del servizio SQL Server disponga dell'accesso necessario. Per istruzioni, vedi Configurare gli account e le autorizzazioni dei servizi Windows sul sito Web di Microsoft. 3. Verifica la connettività di rete per le condivisi oni remote. Per ulteriori informazioni, consulta Accesso ai dati tramite condivisioni di file nella AWS documentazione.	DBA

Automatizza le attività con Task Scheduler

Operazione	Description	Competenze richieste
Crea file batch.	• Per creare un file batch, utilizzate un editor di testo	AWS DevOps, DBA

Operazione	Description	Competenze richieste

	per digitare il seguente comando. Sostituite i parametri `username` e `password` con i vostri valori. Quindi salva il file con nome`<Name>.bat` .
	
	`sqlcmd -S servername -U username -P password -i <T-SQL query path.sql>`
	
	• Per creare un file batch per le attività SQL, utilizzate un editor di testo e digitate i seguenti comandi. Sostituite i parametri `ServerName DatabaseName username`,, e `password` con i vostri valori. Quindi salva il file con nome`<Name>.bat` .
	
	`@echo off`
	`sqlcmd -S [ServerName] -d [DatabaseName] -U username -P password -i "PathToSQLScript\Script.sql" -o "PathToOutput\Output.txt"`	

Operazione	Description	Competenze richieste
Crea attività in Task Scheduler .	Per creare un'attività in Task Scheduler, utilizza i seguenti passaggi: 1. Per aprire Task Scheduler, digitare taskschd.msc nella ricerca di Windows. 2. Scegli il menu Azione, quindi seleziona Crea attività di base. 3. Per Nome, fornisci un nome per l'attività, quindi scegli Avanti. 4. Per Trigger, seleziona l'opzione relativa al momento in cui desideri che l'attività inizi, quindi scegli Avanti. 5. Fornisci le informazioni di inizio e ricorrenza per l'attività, quindi scegli Avanti. 6. Nella sezione Azione, seleziona Avvia un programma, quindi scegli Avanti. 7. Per Programma/script, specificate il percorso del file batch creato nell'oper azione precedente, quindi scegliete Avanti. 8. Scegli Fine.	DBA

Operazione	Description	Competenze richieste
	Per eseguire l'operazione manualmente, fate clic con il pulsante destro del mouse sull'attività appena creata, quindi selezionate Esegui.	
Visualizza lo stato delle attività.	Per visualizzare lo stato di un'attività in Task Scheduler, utilizza i seguenti passaggi: 1. In Task Scheduler, vai alla Task Scheduler Library, che mostra tutte le attività. 2. Per visualizzare lo stato dell'attività creata in precedenza, seleziona l'attività e vai alla scheda Cronologia.	DBA, AWS DevOps

Automatizza le attività con AWS Lambda

Operazione	Description	Competenze richieste
Implementa la soluzione.	Per implementare la soluzione di questo modello, utilizzare i seguenti passaggi: 1. Creazione di una funzione Lambda. Per istruzioni, consulta Creare la prima funzione Lambda nella AWS documentazione. 2. Pianifica la funzione Lambda. Per istruzion i, consulta Invocare una	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	funzione Lambda in base a una pianificazione nella AWS documentazione. 3. Esegui query T-SQL. Per ulteriori informazioni, consulta Tutorial: Using a Lambda function to access a un database Amazon RDS nella documenta zione. AWS Il tutorial spiega come connettere i database Amazon RDS dalle funzioni Lambda per eseguire query SQL.	

risoluzione dei problemi

Problema	Soluzione
Problemi con Lambda	Per informazioni su errori e problemi che potresti riscontrare durante l'utilizzo AWS Lambda, consulta Risoluzione dei problemi in Lambda nella AWS documentazione.

Risorse correlate

- Tipi di EC2 istanze Amazon
- AWS Lambda documentazione
- AWS Lambda prezzi
- Task Scheduler per sviluppatori (sito Web Microsoft)

Automatizza il failover e il failback tra regioni utilizzando DR Orchestrator Framework

Jitendra Kumar, Pavithra Balasubramanian e Oliver Francis, Amazon Web Services

Riepilogo

Questo modello descrive come utilizzare DR Orchestrator Framework per orchestrare e automatizzare i passaggi manuali, soggetti a errori, per eseguire il disaster recovery nelle regioni di Amazon Web Services ().AWS Il modello copre i seguenti database:

- Amazon Relational Database Service (Amazon RDS) per MySQL, Amazon RDS per PostgreSQL o Amazon RDS per MariaDB
- Edizione compatibile con Amazon Aurora MySQL o edizione compatibile con Amazon Aurora PostgreSQL (utilizzando un file centralizzato)
- Amazon ElastiCache (sistema operativo Redis)

Per dimostrare la funzionalità di DR Orchestrator Framework, crei due istanze o cluster DB. Il primario è in e il Regione AWS `us-east-1` secondario è in. `us-west-2` Per creare queste risorse, si utilizzano i AWS CloudFormation modelli nella `App-Stack` cartella del GitHub repository aws-cross-region-dr-databases.

Prerequisiti e limitazioni

Prerequisiti generali

- DR Orchestrator Framework distribuito sia in modalità primaria che secondaria Regioni AWS
- Due bucket Amazon Simple Storage Service
- Un cloud privato virtuale (VPC) con due sottoreti e un gruppo di sicurezza AWS

Prerequisiti specifici del motore

- Amazon Aurora: almeno un database globale di Aurora deve essere disponibile in due. Regioni AWSÈ possibile utilizzare `us-east-1` come regione principale e `us-west-2` come regione secondaria.

- Amazon ElastiCache (Redis OSS): un datastore ElastiCache globale deve essere disponibile in due. Regioni AWS Puoi `use us-east-1` utilizzarla come regione principale e `us-west-2` come regione secondaria.

Limitazioni di Amazon RDS

- DR Orchestrator Framework non verifica il ritardo di replica prima di eseguire un failover o un failback. Il ritardo di replica deve essere controllato manualmente.
- Questa soluzione è stata testata utilizzando un'istanza di database principale con una replica di lettura. Se desideri utilizzare più di una replica di lettura, testa accuratamente la soluzione prima di implementarla in un ambiente di produzione.

Limitazioni di Aurora

- La disponibilità e il supporto delle funzionalità variano a seconda delle versioni specifiche di ciascun motore di database e tra Regioni AWS di loro. Per ulteriori informazioni sulle funzionalità e sulla disponibilità regionale per la replica tra regioni, consulta Repliche di lettura tra regioni.
- I database globali Aurora hanno requisiti di configurazione specifici per le classi di istanze Aurora DB supportate e il numero massimo di. Regioni AWS Per ulteriori informazioni, consulta Requisiti di configurazione di un database globale Amazon Aurora.
- Questa soluzione è stata testata utilizzando un'istanza di database principale con una replica di lettura. Se desideri utilizzare più di una replica di lettura, testa accuratamente la soluzione prima di implementarla in un ambiente di produzione.

ElastiCache limitazioni

- Per informazioni sulla disponibilità regionale per Global Datastore e sui requisiti di ElastiCache configurazione, consulta Prerequisiti e limitazioni nella documentazione. ElastiCache

Versioni del prodotto Amazon RDS p

Amazon RDS supporta le seguenti versioni del motore:

- MySQL — Amazon RDS supporta istanze DB che eseguono le seguenti versioni di MySQL: MySQL 8.0 e MySQL 5.7

- PostgreSQL: per informazioni sulle versioni supportate di Amazon RDS for PostgreSQL, consulta Versioni disponibili del database PostgreSQL.
- MariaDB — Amazon RDS supporta istanze DB che eseguono le seguenti versioni di MariaDB:
  - MariaDB 10.11
  - MariaDB 10.6
  - MariaDB 10.5

Versioni del prodotto Aurora

- Il passaggio al database globale di Amazon Aurora richiede Aurora MySQL, compatibile con MySQL 5.7, versione 2.09.1 e successive

  Per ulteriori informazioni, consulta Limitazioni dei database globali di Amazon Aurora.

ElastiCache Versioni del prodotto (Redis OSS)

Amazon ElastiCache (Redis OSS) supporta le seguenti versioni di Redis:

- Redis 7.1 (avanzata)
- Redis 7.0 (avanzata)
- Redis 6.2 (avanzato)
- Redis 6.0 (avanzato)
- Redis 5.0.6 (potenziato)

Per ulteriori informazioni, consulta Versioni supportate ElastiCache (Redis OSS).

Architecture

Architettura Amazon RDS

L'architettura Amazon RDS include le seguenti risorse:

- L'istanza database Amazon RDS primaria creata nella regione primaria (`us-east-1`) con read/write accesso per i client
- Una replica di lettura Amazon RDS creata nella regione secondaria (`us-west-2`) con accesso in sola lettura per i client

- DR Orchestrator Framework distribuito sia nella regione primaria che in quella secondaria

Il diagramma mostra:

1. Replica asincrona tra l'istanza primaria e l'istanza secondaria
2. Accesso in lettura/scrittura per i client nella regione principale
3. Accesso in sola lettura per i client nella regione secondaria

Architettura Aurora

L'architettura Amazon Aurora include le seguenti risorse:

- Il cluster Aurora DB primario creato nella regione primaria (`us-east-1`) con un endpoint active-writer
- Un cluster Aurora DB creato nella regione secondaria (`us-west-2`) con un endpoint inactive-writer
- DR Orchestrator Framework distribuito sia nella regione primaria che in quella secondaria

Il diagramma mostra:

1. Replica asincrona tra il cluster primario e il cluster secondario
2. Il cluster DB primario con un endpoint active-writer
3. Il cluster DB secondario con un endpoint di scrittura inattiva

ElastiCache Architettura (Redis OSS)

L'architettura Amazon ElastiCache (Redis OSS) include le seguenti risorse:

- Un datastore globale ElastiCache (Redis OSS) creato con due cluster:
  1. Il cluster primario nella regione primaria () `us-east-1`
  2. Il cluster secondario nella regione secondaria (`us-west-2`)
- Un collegamento Amazon interregionale con crittografia TLS 1.2 tra i due cluster
- DR Orchestrator Framework distribuito nelle regioni primarie e secondarie

Automazione e scalabilità

DR Orchestrator Framework è scalabile e supporta il failover o il failback di più di un database in parallelo. AWS

Puoi utilizzare il seguente codice di payload per eseguire il failover di più database del tuo account. AWS In questo esempio, tre AWS database (due database globali come Aurora MySQL compatibile o Aurora PostgreSQL e un'istanza Amazon RDS for MySQL) eseguono il failover nella regione DR:

```
{
  "StatePayload": [
    {
      "layer": 1,
      "resources": [
        {
          "resourceType": "PlannedFailoverAurora",
          "resourceName": "Switchover (planned failover) of Amazon Aurora global
 databases (MySQL)",
          "parameters": {
            "GlobalClusterIdentifier": "!Import dr-globaldb-cluster-mysql-global-
identifier",
            "DBClusterIdentifier": "!Import dr-globaldb-cluster-mysql-cluster-
identifier"
          }
        },
        {
          "resourceType": "PlannedFailoverAurora",
          "resourceName": "Switchover (planned failover) of Amazon Aurora global
 databases (PostgreSQL)",
          "parameters": {
            "GlobalClusterIdentifier": "!Import dr-globaldb-cluster-postgres-global-
identifier",
            "DBClusterIdentifier": "!Import dr-globaldb-cluster-postgres-cluster-
identifier"
          }
        },
        {
          "resourceType": "PromoteRDSReadReplica",
          "resourceName": "Promote RDS for MySQL Read Replica",
          "parameters": {
            "RDSInstanceIdentifier": "!Import rds-mysql-instance-identifier",
```

```
            "TargetClusterIdentifier": "!Import rds-mysql-instance-global-arn"
          }
        }
      ]
    }
  ]
}
```

Tools (Strumenti)

AWS servizi

- Amazon Aurora è un motore di database relazionale completamente gestito creato per il cloud e compatibile con MySQL e PostgreSQL.
- Amazon ti ElastiCache aiuta a configurare, gestire e scalare ambienti di cache in memoria distribuiti in. Cloud AWS Questo modello utilizza Amazon ElastiCache (Redis OSS).
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi. In questo modello, le funzioni Lambda vengono utilizzate AWS Step Functions per eseguire i passaggi.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale in. Cloud AWS Questo modello supporta Amazon RDS for MySQL, Amazon RDS per PostgreSQL e Amazon RDS per MariaDB.
- AWS SDK per Python (Boto3)ti aiuta a integrare la tua applicazione, libreria o script Python con. Servizi AWS In questo modello, i Boto3 APIs vengono utilizzati per comunicare con le istanze del database o i database globali.
- AWS Step Functionsè un servizio di orchestrazione senza server che consente di combinare AWS Lambda funzioni e altro per creare applicazioni aziendali critiche. Servizi AWS In questo modello, le macchine a stati Step Functions vengono utilizzate per orchestrare ed eseguire il failover e il failback interregionali delle istanze del database o dei database globali.

Archivio di codice

Il codice per questo pattern è disponibile nel repository aws-cross-region-dr-databases su. GitHub

Epiche

Installa DR Orchestrator Framework

Operazione	Description	Competenze richieste
Clona il GitHub repository.	Per clonare il repository, esegui il seguente comando: `git clone https://github.com/aws-samples/aws-cross-region-dr-databases.git`	AWS DevOps, amministratore AWS

Codice delle funzioni Package Lambda in un archivio di file.zip.	Crea i file di archivio per le funzioni Lambda per includere le dipendenze di DR Orchestrator Framework:
	`cd <YOUR-LOCAL-GIT-FOLDER>/DR-Orchestration-artifacts`
	
	`bash scripts/deploy-orchestrator-sh.sh`
		Amministratore AWS

Crea bucket S3.	I bucket S3 sono necessari per archiviare DR Orchestrator Framework insieme alla configurazione più recente. Crea due bucket S3, uno nella regione principale () e uno nella regione secondaria (`us-east-1`): `us-west-2` • `dr-orchestrator-xxxxxx-us-east-1`	Amministratore AWS

Operazione	Description	Competenze richieste
	• `dr-orchestrator-xxxxxx-us-west-2` `xxxxxx`Sostituiscili con un valore casuale per rendere unici i nomi dei bucket.	
Crea sottoreti e gruppi di sicurezza.	Sia nella regione principale (`us-east-1`) che nella regione secondaria (`us-west-2`), crea due sottoreti e un gruppo di sicurezza per l'implementazione della funzione Lambda nel tuo VPC: • `subnet-XXXXXXX` • `subnet-YYYYYYY` • `sg-XXXXXXXXXXX`	Amministratore AWS

Operazione	Description	Competenze richieste
Aggiorna i file dei parametri di DR Orchestrator.	Nella <YOUR-LOCAL-GIT-FOLDER>/DR-Orchestration-artifacts/cloudformation cartella, aggiorna i seguenti file dei parametri di DR Orchestrator: • Orchestrator-Deployer-parameters-us-east-1.json • Orchestrator-Deployer-parameters-us-west-2.json Utilizza i seguenti valori dei parametri, sostituendo x e y con i nomi delle tue risorse: [{ "ParameterKey": "TemplateStoreS3BucketName", "ParameterValue": "dr-orchestrator-xxxxxx-us-east-1" }, { "ParameterKey": "TemplateVPCId", "ParameterValue": "vpc-xxxxxx" }, {	Amministratore AWS

Operazione	Description	Competenze richieste
	"ParameterKey": "TemplateLambdaSubnetID1", "ParameterValue": "subnet-xxxxx" }, { "ParameterKey": "TemplateLambdaSubnetID2", "ParameterValue": "subnet-yyyyy" }, { "ParameterKey": "TemplateLambdaSecurityGroupID", "ParameterValue": "sg-xxxxxxxxxx" }]	

Operazione	Description	Competenze richieste
Carica il codice di DR Orchestrator Framework nel bucket S3.	Il codice sarà più sicuro in un bucket S3 che nella directory locale. Carica la `DR-Orchestration-artifacts` directory, inclusi tutti i file e le sottocartelle, nei bucket S3. Per caricare il codice, procedi come segue: 1. Accedi alla Console di gestione AWS. 2. Accedi alla console Amazon S3. 3. Selezionare il `dr-orchestrator-xxxxxx-us-east-1 bucket` . 4. Scegli Carica, quindi scegli Aggiungi cartella. 5. Selezionare la cartella `DR-Orchestration-artifacts` . 6. Scegli Carica. 7. Seleziona il `dr-orchestrator-xxxxxx-us-west-2` secchio. 8. Ripetere i passaggi 4-7.	Amministratore AWS

Operazione	Description	Competenze richieste

Implementa DR Orchestrator Framework nella regione principale.	Per distribuire DR Orchestrator Framework nella regione primaria (), esegui i seguenti comandi: `us-east-1`
	
	`cd <YOUR-LOCAL-GIT-FOLDER>/DR-Orchestration-artifacts/cloudformation`
	
	`aws cloudformation deploy \`
	`--region us-east-1 \`
	`--stack-name dr-orchestrator \`
	`--template-file Orchestrator-Deployer.yaml \`
	`--parameter-overrides file://Orchestrator-Deployer-parameters-us-east-1.json \`
	`--capabilities CAPABILITY_AUTO_EXPAND CAPABILITY_NAMED_IAM CAPABILITY_IAM \`
	`--disable-rollback`
		Amministratore AWS

Operazione	Description	Competenze richieste

Implementa DR Orchestrator Framework nella regione secondaria.	Nella regione secondaria (`us-west-2`), esegui i seguenti comandi:
	`cd <YOUR-LOCAL-GIT-FOLDER>/DR-Orchestration-artifacts/cloudformation`
	
	`aws cloudformation deploy \`
	`--region us-west-2 \`
	`--stack-name dr-orchestrator \`
	`--template-file Orchestrator-Deployer.yaml \`
	`--parameter-overrides file://Orchestrator-Deployer-parameters-us-west-2.json \`
	`--capabilities CAPABILITY_AUTO_EXPAND CAPABILITY_NAMED_IAM CAPABILITY_IAM \`
	`--disable-rollback`
		Amministratore AWS

Operazione	Description	Competenze richieste
Verifica la distribuzione.	Se il CloudFormation comando viene eseguito correttamente, restituisce il seguente risultato: `Successfully created/ updated stack - dr-orchestrator` In alternativa, puoi accedere alla CloudFormation console e verificare lo stato dello `dr-orchestrator` stack.	Amministratore AWS

Crea le istanze o i cluster del database

Operazione	Description	Competenze richieste
Creare le sottoreti e i gruppi di sicurezza del database.	Nel tuo VPC, crea due sottoreti e un gruppo di sicurezza per l'istanza DB o il database globale nelle regioni primaria (`us-east-1`) e secondaria (): `us-west-2` • `subnet-XXXXXX` • `subnet-XXXXXX` • `sg-XXXXXXXXXX`	Amministratore AWS
Aggiorna il file dei parametri per l'istanza o il cluster DB primario.	Nella `<YOUR LOCAL GIT FOLDER>/App-Stack` cartella, aggiorna il file dei parametri per la regione principale.	Amministratore AWS

Operazione	Description	Competenze richieste
	Amazon RDS Nel `RDS-MySQL-parameter-us-east-1.json` file, aggiorna `SubnetIds` e `DBSecurityGroup` inserisci i nomi delle risorse che hai creato: `{ "Parameters": { "SubnetIds": "subnet-xxxxxx,subnet-xxxxxx", "DBSecurityGroup": "sg-xxxxxxxxxx", "MySqlGlobalIdentifier":"rds-mysql-instance", "InitialDatabaseName": "mysqldb", "DBPortNumber": "3789", "PrimaryRegion": "us-east-1", "SecondaryRegion": "us-west-2", "KMSKeyAliasName": "rds/rds-mysql-instance-KmsKeyId" } }`  Amazon Aurora  Nel `Aurora-MySQL-parameter-us-east-1.json` file, aggiorna `SubnetIds` e `DBSecurityGroup` con	

Operazione	Description	Competenze richieste
	i nomi delle risorse che hai creato: `{ "Parameters": { "SubnetIds": "subnet1-xxxxxx,subnet2-xxxxxx", "DBSecurityGroup": "sg-xxxxxxxxxx", "GlobalClusterIdentifier":"dr-globaldb-cluster-mysql", "DBClusterName":"dbcluster-01", "SourceDBClusterName":"dbcluster-02", "DBPortNumber": "3787", "DBInstanceClass": "db.r5.large", "InitialDatabaseName": "sampledb", "PrimaryRegion": "us-east-1", "SecondaryRegion": "us-west-2", "KMSKeyAliasName": "rds/dr-globaldb-cluster-mysql-KmsKeyId" } }`  Amazon ElastiCache (sistema operativo Redis)  Nel `ElastiCache-parameter-us-east-1.json` file, aggiorna `SubnetIds` e	

Operazione	Description	Competenze richieste
	`DBSecurityGroup` inserisci i nomi delle risorse che hai creato. `{ "Parameters": { "CacheNodeType": "cache.m5.large", "DBSecurityGroup": "sg-xxxxxxxxxx", "SubnetIds": "subnet-xxxxxx,subnet-xxxxxx", "EngineVersion": "5.0.6", "GlobalReplicationGroupIdSuffix": "demo-redis-global-datastore", "NumReplicas": "1", "NumShards": "1", "ReplicationGroupId": "demo-redis-cluster", "DBPortNumber": "3788", "TransitEncryption": "true", "KMSKeyAliasName": "elasticache/demo-redis-global-datastore-KmsKeyId", "PrimaryRegion": "us-east-1", "SecondaryRegion": "us-west-2" } }`	

Operazione	Description	Competenze richieste

Distribuisci l'istanza o il cluster di database nella regione principale.	Per distribuire l'istanza o il cluster nella regione primaria (`us-east-1`), esegui i seguenti comandi in base al motore di database.
	
	Amazon RDS
	
	`cd <YOUR-LOCAL-GIT-FOLDER>/App-Stack`
	
	`aws cloudformation deploy \`
	`--region us-east-1 \`
	`--stack-name rds-mysql-app-stack \`
	`--template-file RDS-MySQL-Primary.yaml \`
	`--parameter-overrides file://RDS-MySQL-parameter-us-east-1.json \`
	`--capabilities CAPABILITY_AUTO_EXPAND CAPABILITY_NAMED_IAM CAPABILITY_IAM \`
	`--disable-rollback`
	
	Amazon Aurora
	
	`cd <YOUR-LOCAL-GIT-FOLDER>/App-Stack`
	
	`aws cloudformation deploy \`
	`--region us-east-1 \`
	`--stack-name aurora-mysql-app-stack \`
		Amministratore AWS

Operazione	Description	Competenze richieste

	`--template-file Aurora-MySQL-Primary.yaml \ --parameter-overrides file://Aurora-MySQL-parameter-us-east-1.json \ --capabilities CAPABILITY_AUTO_EXPAND CAPABILITY_NAMED_IAM CAPABILITY_IAM \ --disable-rollback`
	
	Amazon ElastiCache (sistema operativo Redis)
	
	`cd <YOUR-LOCAL-GIT-FOLDER>/App-Stack`
	
	`aws cloudformation deploy \ --region us-east-1 --stack-name elasticache-ds-app-stack \ --template-file ElastiCache-Primary.yaml \ --parameter-overrides file://ElastiCache-parameter-us-east-1.json \ --capabilities CAPABILITY_AUTO_EXPAND CAPABILITY_NAMED_IAM CAPABILITY_IAM \ --disable-rollback`	

Operazione	Description	Competenze richieste
	Verifica che le CloudFormation risorse siano state distribuite correttamente.	

Operazione	Description	Competenze richieste
Aggiorna il file dei parametri per l'istanza o il cluster DB secondario.	Nella `<YOUR LOCAL GIT FOLDER>/App-Stack` cartella, aggiorna il file dei parametri per la regione secondaria. Amazon RDS Nel `RDS-MySQL-parameter-us-west-2.json` file, aggiorna `SubnetIDs` e `DBSecurityGroup` inserisci i nomi delle risorse che hai creato. Aggiorna il file `PrimaryRegionKMSKeyArn` con il valore `MySQLKmsKeyId` preso dalla sezione Outputs dello CloudFormation stack per l'istanza DB principale: `{ "Parameters": { "SubnetIds": "subnet-aaaaaaaaa, subnet-bbbbbbbbb", "DBSecurityGroup": "sg-ccccccccc", "MySqlGlobalIdentifier":"rds-mysql-instance", "InitialDatabaseName": "mysqldb", "DBPortNumber": "3789", "PrimaryRegion": "us-east-1",`	Amministratore AWS

Operazione	Description	Competenze richieste

	<pre> "SecondaryRegion":
	 "us-west-2",
	    "KMSKeyAliasName":
	 "rds/rds-mysql-ins
	tance-KmsKeyId",
	    "PrimaryRegionKMSK
	eyArn":"arn:aws:km
	s:us-east-1:xxxxxx
	xxx:key/mrk-xxxxxx
	xxxxxxxxxxxxxx"
	  }
	}</pre>
	Amazon Aurora
	
	Nel `Aurora-MySQL-param eter-us-west-2.jso n` file, aggiorna `SubnetIDs` e inserisci `DBSecurit yGroup` i nomi delle risorse che hai creato. Aggiorna il `PrimaryRe gionKMSKeyArn` con il valore `AuroraKmsKeyId` preso dalla sezione Outputs dello CloudFormation stack per l'istanza DB principale:
	
	<pre>{
	  "Parameters": {
	    "SubnetIds":
	 "subnet1-aaaaaaaaa
	,subnet2-bbbbbbbbb",
	    "DBSecurityGroup":
	 "sg-ccccccccc",
	    "GlobalClusterIden
	tifier":"dr-globaldb-
	cluster-mysql",</pre>	

Operazione	Description	Competenze richieste

	<pre>"DBClusterName":"dbcluster-01",
	"SourceDBClusterName":"dbcluster-02",
	"DBPortNumber": "3787",
	"DBInstanceClass": "db.r5.large",
	"InitialDatabaseName": "sampledb",
	"PrimaryRegion": "us-east-1",
	"SecondaryRegion": "us-west-2",
	"KMSKeyAliasName": "rds/dr-globaldb-cluster-mysql-KmsKeyId"
	}
	}</pre> Amazon ElastiCache (sistema operativo Redis)
	
	Nel `ElastiCache-parameter-us-west-2.json` file, aggiorna `SubnetIDs` e `DBSecurityGroup` inserisci i nomi delle risorse che hai creato. Aggiorna il file `PrimaryRegionKMSKeyArn` con il valore `ElastiCacheKmsKeyId` preso dalla sezione Outputs dello CloudFormation stack per l'istanza DB principale:
	
	<pre>{
	"Parameters": {</pre>	

Operazione	Description	Competenze richieste
	"CacheNodeType": "cache.m5.large", "DBSecurityGroup": "sg-ccccccccc", "SubnetIds": "subnet-aaaaaaaaa, subnet-bbbbbbbbb", "EngineVersion": "5.0.6", "GlobalReplicationGroupIdSuffix": "demo-redis-global-datastore", "NumReplicas": "1", "NumShards": "1", "ReplicationGroupId": "demo-redis-cluster", "DBPortNumber": "3788", "TransitEncryption": "true", "KMSKeyAliasName": "elasticache/demo-redis-global-datastore-KmsKeyId", "PrimaryRegion": "us-east-1", "SecondaryRegion": "us-west-2" } }	

Operazione	Description	Competenze richieste

Distribuisci l'istanza o il cluster di database nella regione secondaria.	Esegui i seguenti comandi, in base al tuo motore di database.
	
	Amazon RDS
	
	`cd <YOUR-LOCAL-GIT-FOLDER>/App-Stack`
	
	`aws cloudformation deploy \`
	`--region us-west-2 \`
	`--stack-name rds-mysql-app-stack \`
	`--template-file RDS-MySQL-DR.yaml \`
	`--parameter-overrides file://RDS-MySQL-parameter-us-west-2.json \`
	`--capabilities CAPABILITY_AUTO_EXPAND CAPABILITY_NAMED_IAM CAPABILITY_IAM \`
	`--disable-rollback`
	
	Amazon Aurora
	
	`cd <YOUR-LOCAL-GIT-FOLDER>/App-Stack`
	
	`aws cloudformation deploy \`
	`--region us-west-2 \`
	`--stack-name aurora-mysql-app-stack \`
	`--template-file Aurora-MySQL-DR.yaml \`
		Amministratore AWS

Operazione	Description	Competenze richieste

	`--parameter-overrides file://Aurora-MySQL-parameter-us-west-2.json \ --capabilities CAPABILITY_AUTO_EXPAND CAPABILITY_NAMED_IAM CAPABILITY_IAM \ --disable-rollback`
	
	Amazon ElastiCache (sistema operativo Redis)
	
	`cd <YOUR-LOCAL-GIT-FOLDER>/App-Stack`
	
	`aws cloudformation deploy \ --region us-west-2 \ --stack-name elasticache-ds-app-stack \ --template-file ElastiCache-DR.yaml \ --parameter-overrides file://ElastiCache-parameter-us-west-2.json \ --capabilities CAPABILITY_AUTO_EXPAND CAPABILITY_NAMED_IAM CAPABILITY_IAM \ --disable-rollback`
	
	Verifica che le CloudFormation risorse siano state distribuite correttamente.	

Risorse correlate

- Strategia di disaster recovery per i database su AWS(strategiaAWS Prescriptive Guidance)
- Automatizza la tua soluzione DR per i database relazionali su AWS(guida Prescriptive Guidance)AWS
- Using Amazon Aurora global databases
- Replica attraverso l'utilizzo di datastore globali Regioni AWS
- Automatizza la tua soluzione DR per i database relazionali su AWS (guida Prescriptive Guidance)AWS

Esegui automaticamente il backup dei database SAP HANA utilizzando Systems Manager e EventBridge

Ambarish Satarkar e Gaurav Rath, Amazon Web Services

Riepilogo

Questo modello descrive come automatizzare i backup dei database SAP HANA utilizzando AWS Systems Manager, Amazon, EventBridge Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) e AWS Backint Agent per SAP HANA.

Questo modello fornisce un approccio basato su shell script che utilizza il `BACKUP DATA` comando ed elimina la necessità di mantenere script e configurazioni di lavoro per ogni istanza del sistema operativo (OS) su numerosi sistemi.

> Nota: ad aprile 2023, AWS Backup ha annunciato il supporto per i database SAP HANA su Amazon Elastic Compute Cloud (Amazon). EC2 Per ulteriori informazioni, consulta Database SAP HANA sul backup delle EC2 istanze Amazon.
>
> In base alle esigenze della tua organizzazione, puoi utilizzare il servizio AWS Backup per eseguire automaticamente il backup dei database SAP HANA oppure puoi utilizzare questo schema.

Prerequisiti e limitazioni

Prerequisiti

- Un'istanza SAP HANA esistente con una versione supportata in stato di esecuzione su un'istanza Amazon Elastic Compute Cloud (Amazon EC2) gestita configurata per Systems Manager
- Systems Manager Agent (SSM Agent) 2.3.274.0 o versione successiva installata
- Un bucket S3 per cui non è abilitato l'accesso pubblico
- Una chiave denominata `hdbuserstore SYSTEM`
- Un ruolo AWS Identity and Access Management (IAM) per il runbook di automazione da eseguire nei tempi previsti
- `AmazonSSMManagedInstanceCoree ssm:StartAutomationExecution` le policy sono associate al ruolo del servizio Systems Manager Automation.

Limitazioni

- AWS Backint Agent per SAP HANA non supporta la deduplicazione.
- AWS Backint Agent per SAP HANA non supporta la compressione dei dati.

Versioni del prodotto

AWS Backint Agent è supportato sui seguenti sistemi operativi:

- SUSE Linux Enterprise Server
- SUSE Linux Enterprise Server per SAP
- Red Hat Enterprise Linux per SAP

AWS Backint Agent supporta i seguenti database:

- SAP HANA 1.0 SP12 (nodo singolo e nodi multipli)
- SAP HANA 2.0 e versioni successive (nodo singolo e nodi multipli)

Architecture

Stack tecnologico Target

- Agente AWS Backint
- Simple Storage Service (Amazon S3)
- AWS Systems Manager
- Amazon EventBridge
- SAP HANA

Architettura Target

Il diagramma seguente mostra gli script di installazione che installano AWS Backint Agent, il bucket S3 e Systems EventBridge Manager e che utilizzano un documento Command per pianificare backup regolari.

Automazione e scalabilità

- È possibile installare più agenti AWS Backint utilizzando un runbook Systems Manager Automation.
- Ogni esecuzione del runbook Systems Manager può essere scalata fino a un numero n di istanze SAP HANA, in base alla selezione del target.
- EventBridge può automatizzare i backup SAP HANA.

Tools (Strumenti)

- AWS Backint Agent per SAP HANA è un'applicazione autonoma che si integra con i flussi di lavoro esistenti per eseguire il backup del database SAP HANA in un bucket S3 specificato nel file di configurazione. AWS Backint Agent supporta backup completi, incrementali e differenziali dei database SAP HANA. Funziona su un server di database SAP HANA, dove i backup e i cataloghi vengono trasferiti dal database SAP HANA all'AWS Backint Agent.
- Amazon EventBridge è un servizio di bus eventi senza server che puoi utilizzare per connettere le tue applicazioni con dati provenienti da una varietà di fonti. EventBridge fornisce un flusso di dati in tempo reale dalle tue applicazioni, dalle applicazioni SaaS (SaaS) e dai servizi AWS a target come funzioni AWS Lambda, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altri account.
- Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è un servizio di storage di oggetti. È possibile utilizzare Amazon S3 per memorizzare e recuperare qualsiasi volume di dati, in qualunque momento e da qualunque luogo tramite il Web.
- AWS Systems Manager ti aiuta a visualizzare e controllare la tua infrastruttura su AWS. Utilizzando la console Systems Manager, puoi visualizzare i dati operativi da più servizi AWS e automatizzare le attività operative tra le tue risorse AWS.

Codice

Il codice per questo pattern è disponibile nel aws-backint-automated-backup GitHub repository.

Epiche

Crea un sistema di chiavi hdbuserstore

Operazione	Description	Competenze richieste

Crea una chiave hdbuserstore.	1. Accedi a `/usr/sap/<SID>/HDB<Inst No>/exe`.
	2. Esegui il comando seguente, con XX come numero di istanza del database SAP HANA.
	`hdbuserstore -i set SYSTEM <hostname>:3XX13@SYSTEMDB SYSTEM`
	Ad esempio, per un host SAP HANA `saphanadb` con numero di istanza`00`, esegui il comando seguente.
	`hdbuserstore -i set SYSTEM saphanadb:30013@SYSTEMDB SYSTEM`
		Amministratore AWS, amministratore SAP HANA

Installa AWS Backint Agent

Operazione	Description	Competenze richieste
Installa AWS Backint Agent.	Segui le istruzioni in Installare e configurare AWS Backint Agent per SAP HANA nella	Amministratore AWS, amministratore SAP HANA

Operazione	Description	Competenze richieste
	documentazione di AWS Backint Agent.	

Creare il documento Systems Manager Command

Operazione	Description	Competenze richieste
Creare il documento Systems Manager Command.	1. Accedi alla Console di gestione AWS e apri la console AWS Systems Manager. 2. Scegli Documents e scegli Owned by me. 3. Verifica di trovarti nella stessa regione AWS del tuo database SAP HANA. 4. Scegli Crea documento, Comando o sessione per creare il tuo documento. 5. Usa un nome univoco e descrittivo, senza spazi (ad esempio, SAP HANA-Back up). 6. Assicurati che il tipo di documento sia impostato su Documento di comando. 7. Sotto l'intestazione Content, c'è un codice di esempio. Assicurati di scegliere il tipo di codice JSON e sostituisci il codice con il codice contenuto nel `HDB_Backup_SSM_Document.jso`	Amministratore AWS, amministratore SAP HANA

Operazione	Description	Competenze richieste
	n file del repository. GitHub 8. Scegli Crea documento. 9. Controlla il tuo documento nella sezione Owned by me.	

Pianifica i backup con una frequenza regolare

Operazione	Description	Competenze richieste
Pianifica backup regolari con Amazon EventBridge.	1. Apri la EventBridge console Amazon, scegli Regole e scegli Crea regola. 2. Nella schermata Definisci i dettagli della regola, inserisci un nome e una descrizione univoci per la regola e utilizza il bus di eventi predefinito. 3. In Tipo di regola, scegli Pianifica e scegli Avanti. 4. Nella schermata Definisci pianificazione, scegli lo schema di pianificazione appropriato e l'espress ione cron o rate appropria ta in base alla frequenza richiesta. 5. Nella schermata Seleziona obiettivi, per Tipo di destinazione, scegli il servizio AWS. In Seleziona	Amministratore AWS, amministratore SAP HANA

Operazione	Description	Competenze richieste
	una destinazione, scegli Systems Manager Run Command. 6. Scegliete il documento che avete creato in precedenza. 7. In Target key e Target value, fornisci l'ID dell'istanza. Puoi utilizzare i nomi e i valori dei tag per aggiungere più istanze. 8. In Configura i parametri di automazione, scegli Constante per i backup incrementali o differenziali. Se desideri un backup completo, scegli Nessun parametro. 9. Scegli se creare un nuovo ruolo o utilizzare un ruolo esistente. Se utilizzi un ruolo esistente, assicurati che abbia le politiche necessarie per richiamare l'obiettivo. 10. Mantieni le impostazioni aggiuntive predefinite e scegli Avanti. 11. La schermata Configura tag è facoltativa. Scegli Avanti. 12. Nella schermata Rivedi e crea, rivedi le impostazioni della regola e scegli Crea.	

Operazione	Description	Competenze richieste
	La regola deve essere creata correttamente. Puoi verificare il successo del backup dal percorso del bucket S3. `s3:/<your_bucket_name>/<target folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/backint/DB_<SID>/` Puoi anche verificare i backup dal catalogo di backup SAP HANA.	

Risorse correlate

- AWS Backint Agent per SAP HANA
- Installa e configura AWS Backint Agent per SAP HANA

Genera automaticamente un modello PynamoDB e funzioni CRUD per Amazon DynamoDB utilizzando un'applicazione Python

Vijit Vashishtha, Dheeraj Alimchandani e Dhananjay Karanjkar, Amazon Web Services

Riepilogo

È comune richiedere entità e funzioni operative di creazione, lettura, aggiornamento ed eliminazione (CRUD) per eseguire in modo efficiente le operazioni del database Amazon DynamoDB. PynamoDB è un'interfaccia basata su Python che supporta Python 3. Fornisce inoltre funzionalità come il supporto per le transazioni Amazon DynamoDB, la serializzazione e la deserializzazione automatiche dei valori degli attributi e la compatibilità con i framework Python più comuni, come Flask e Django. Questo modello aiuta gli sviluppatori a lavorare con Python e DynamoDB fornendo una libreria che semplifica la creazione automatica di modelli PynamoDB e funzioni operative CRUD. Oltre a generare funzioni CRUD essenziali per le tabelle del database, può anche decodificare i modelli PynamoDB e le funzioni CRUD dalle tabelle Amazon DynamoDB. Questo modello è progettato per semplificare le operazioni del database utilizzando un'applicazione basata su Python.

Le caratteristiche principali di questa soluzione sono le seguenti:

- Da schema JSON a modello PynamoDB: genera automaticamente modelli PynamoDB in Python importando un file di schema JSON.
- Generazione di funzioni CRUD: genera automaticamente funzioni per eseguire operazioni CRUD sulle tabelle DynamoDB.
- Reverse engineering da DynamoDB: utilizza la mappatura relazionale degli oggetti (ORM) di PynamoDB per decodificare i modelli PynamoDB e le funzioni CRUD per le tabelle Amazon DynamoDB esistenti.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Python versione 3.8 o successiva, scaricata e installata
- Jinja2 versione 3.1.2 o successiva, scaricato e installato
- Tabelle Amazon DynamoDB per le quali desideri generare ORM
- AWS Command Line Interface (AWS CLI), installato e configurato

- PynamoDB versione 5.4.1 o successiva, installato

Architecture

Stack tecnologico Target

- Script JSON
- Applicazione Python
- Modello PynamoDB
- Istanza di database Amazon DynamoDB

Architettura di destinazione

1. Si crea un file di schema JSON di input. Questo file di schema JSON rappresenta gli attributi delle rispettive tabelle DynamoDB da cui si desidera creare i modelli PynamoDB e le funzioni CRUD. Contiene le seguenti tre chiavi importanti:
   - `name`—Il nome della tabella DynamoDB di destinazione.
   - `region`— Il Regione AWS luogo in cui è ospitata la tabella
   - `attributes`— Gli attributi che fanno parte della tabella di destinazione, come la chiave di partizione (nota anche come attributo hash), la chiave di ordinamento, gli indici secondari locali, gli indici secondariglobali e tutti gli attributi non chiave. Questo strumento prevede che lo schema di input fornisca solo gli attributi non chiave poiché l'applicazione recupera gli attributi chiave direttamente dalla tabella di destinazione. Per un esempio di come specificare gli attributi nel file di schema JSON, consultate la sezione Informazioni aggiuntive di questo modello.
2. Esegui l'applicazione Python e fornisci il file di schema JSON come input.
3. L'applicazione Python legge il file di schema JSON.
4. L'applicazione Python si connette alle tabelle DynamoDB per derivare lo schema e i tipi di dati. L'applicazione esegue l'operazione describe_table e recupera gli attributi chiave e indice per la tabella.
5. L'applicazione Python combina gli attributi del file di schema JSON e della tabella DynamoDB. Utilizza il motore di template Jinja per generare un modello Pynamodb e le funzioni CRUD corrispondenti.
6. Si accede al modello PynamoDB per eseguire operazioni CRUD sulla tabella DynamoDB.

Tools (Strumenti)

Servizi AWS

- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.

Altri strumenti

- Jinja è un motore di template estensibile che compila i modelli in codice Python ottimizzato. Questo modello utilizza Jinja per generare contenuti dinamici incorporando segnaposto e logica nei modelli.
- PynamoDB è un'interfaccia basata su Python per Amazon DynamoDB.
- Python è un linguaggio di programmazione per computer generico.

Deposito di codice

Il codice per questo modello è disponibile nel repository di modelli PynamoDB a GitHub generazione automatica e funzioni CRUD. Il repository è diviso in due parti principali: il pacchetto controller e i modelli.

Pacchetto controller

Il pacchetto Python del controller contiene la logica dell'applicazione principale che aiuta a generare il modello PynamoDB e le funzioni CRUD. Contiene i seguenti dati:

- `input_json_validator.py`— Questi script Python convalidano il file di schema JSON di input e creano gli oggetti Python che contengono l'elenco delle tabelle DynamoDB di destinazione e gli attributi richiesti per ciascuna di esse.
- `dynamo_connection.py`— Questo script stabilisce una connessione alla tabella DynamoDB e utilizza `describe_table` l'operazione per estrarre gli attributi necessari per creare il modello PynamoDB.
- `generate_model.py`— Questo script contiene una classe Python `GenerateModel` che crea il modello PynamoDB basato sul file di schema JSON di input e sull'operazione. `describe_table`
- `generate_crud.py`— Per le tabelle DynamoDB definite nel file di schema JSON, questo script utilizza l'operazione per creare `GenerateCrud` le classi Python.

Modelli

Questa directory Python contiene i seguenti modelli Jinja:

- `model.jinja`— Questo modello Jinja contiene l'espressione del modello per generare lo script del modello PynamoDB.
- `crud.jinja`— Questo modello Jinja contiene l'espressione del modello per la generazione dello script delle funzioni CRUD.

Poemi epici

Configura l'ambiente

Operazione	Description	Competenze richieste
Clonare il repository.	Immettere il seguente comando per clonare il repository di modelli PynamoDB e funzioni CRUD a generazione automatica. `git clone https://github.com/aws-samples/amazon-reverse-engineer-dynamodb.git`	Sviluppatore di app
Configura l'ambiente Python.	1. Naviga nella directory di primo livello nel repository clonato. `cd amazon-reverse-engineer-dynamodb` 2. Immettete il seguente comando per installare le librerie e i pacchetti richiesti .	Sviluppatore di app

Operazione	Description	Competenze richieste
	`pip install -r requirements.txt`	

Genera il modello PynamoDB e le funzioni CRUD

Operazione	Description	Competenze richieste
Modifica il file dello schema JSON.	1. Naviga nella directory di primo livello del repository clonato. `cd amazon-reverse-engineer-dynamodb` 2. Apri il `test.json` file nel tuo editor preferito. Puoi usare questo file come riferimento per creare il tuo file di schema JSON oppure puoi aggiornare i valori in questo file in modo che corrispondano al tuo ambiente. 3. Modifica il nome e i valori degli attributi per le tabelle DynamoDB di destinazione. Regione AWS **Note** Se si definisce una tabella che non esiste nel file di schema JSON,	Sviluppatore di app

Operazione	Description	Competenze richieste
	questa soluzione non genera modelli o funzioni CRUD per quella tabella. 4. Salvare e chiudere il file `test.json` . Si consiglia di salvare questo file con un nuovo nome.	
Esegui l'applicazione Python.	Inserisci il seguente comando per generare i modelli Pynamodb e le funzioni CRUD, `<input_schema.json>` dov'è il nome del tuo file di schema JSON. `python main.py --file <input_schema.json>`	Sviluppatore di app

Verifica il modello PynamoDB e le funzioni CRUD

Operazione	Description	Competenze richieste
Verifica il modello Pynamodb generato.	1. Nella directory di primo livello del repository clonato, immettete il seguente comando per navigare nel repository. `models` `cd models` 2. Per impostazione predefinita, questa soluzione nomina	Sviluppatore di app

Operazione	Description	Competenze richieste
	il file di modello Pynamodb. `demo_model.py` Verifica che questo file sia presente.	
Verifica le funzioni CRUD generate.	1. Nella directory di primo livello del repository clonato, inserisci il seguente comando per navigare nel repository. `crud` `cd crud` 2. Per impostazione predefini ta, questa soluzione assegna un nome allo script. `demo_crud.py` Verifica che questo file sia presente. 3. Utilizzate le classi Python nel `demo_crud.py` file per eseguire un'operaz ione CRUD sulla tabella DynamoDB di destinazi one. Verificate che l'operazi one sia stata completata correttamente.	Sviluppatore di app

Risorse correlate

- Componenti principali di Amazon DynamoDB (documentazione DynamoDB)
- Miglioramento dell'accesso ai dati con indici secondari (documentazione DynamoDB)

Informazioni aggiuntive

Attributi di esempio per il file di schema JSON

```
[
{
"name": "test_table",
"region": "ap-south-1",
"attributes": [
{
"name": "id",
"type": "UnicodeAttribute"
},
{
"name": "name",
"type": "UnicodeAttribute"
},
{
"name": "age",
"type": "NumberAttribute"
}
]
}
]
```

Blocca l'accesso pubblico ad Amazon RDS utilizzando Cloud Custodian

abhay kumar e Dwarika Patra, Amazon Web Services

Riepilogo

Molte organizzazioni gestiscono i propri carichi di lavoro e servizi su più fornitori di cloud. In questi ambienti cloud ibridi, l'infrastruttura cloud richiede una rigida governance del cloud, oltre alla sicurezza fornita dai singoli provider di cloud. Un database cloud come Amazon Relational Database Service (Amazon RDS) è un servizio importante che deve essere monitorato per eventuali vulnerabilità di accesso e autorizzazione. Sebbene sia possibile limitare l'accesso al database Amazon RDS configurando un gruppo di sicurezza, è possibile aggiungere un secondo livello di protezione per vietare azioni come l'accesso pubblico. Il blocco dell'accesso pubblico ti aiuta a rispettare il Regolamento generale sulla protezione dei dati (GDPR), l'Health Insurance Portability and Accountability Act (HIPAA), il National Institute of Standards and Technology (NIST) e il Payment Card Industry Data Security Standard (PCI DSS).

Cloud Custodian è un motore di regole open source che puoi utilizzare per imporre restrizioni di accesso a risorse Amazon Web Services (AWS) come Amazon RDS. Con Cloud Custodian, puoi impostare regole che convalidano l'ambiente rispetto a standard di sicurezza e conformità definiti. Puoi utilizzare Cloud Custodian per gestire i tuoi ambienti cloud contribuendo a garantire la conformità con le politiche di sicurezza, le politiche di tag e la raccolta dei rifiuti di risorse inutilizzate e la gestione dei costi. Con Cloud Custodian, puoi utilizzare un'unica interfaccia per implementare la governance in un ambiente cloud ibrido. Ad esempio, puoi usare l'interfaccia Cloud Custodian per interagire con e con AWS Microsoft Azure, riducendo lo sforzo di lavorare con meccanismi come gruppi di AWS sicurezza e AWS Config policy di Azure.

Questo modello fornisce istruzioni per utilizzare Cloud Custodian on per AWS imporre la restrizione dell'accessibilità pubblica sulle istanze Amazon RDS.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Una key pair
- AWS Lambda installato

Architecture

Il diagramma seguente mostra Cloud Custodian che implementa la policy su AWS Lambda, AWS CloudTrail avvia l'evento e l'impostazione della `CreateDBInstance` funzione Lambda su false su Amazon RDS. `PubliclyAccessible`

Tools (Strumenti)

Servizi AWS

- AWS CloudTrailti aiuta a verificare la governance, la conformità e il rischio operativo del tuo. Account AWS
- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale in. Cloud AWS

Altri strumenti

- Cloud Custodian unifica gli strumenti e gli script utilizzati da molte organizzazioni per gestire i propri account cloud pubblici in un unico strumento open source. Utilizza un motore di regole stateless per la definizione e l'applicazione delle politiche, con metriche, output strutturati e report dettagliati per l'infrastruttura cloud. Si integra perfettamente con un runtime serverless per fornire correzioni e risposte in tempo reale con un basso sovraccarico operativo.

Epiche

Configura il AWS CLI

Operazione	Description	Competenze richieste
Installa il AWS CLI.	Per installare il AWS CLI, segui le istruzioni nella AWS documentazione.	Amministratore AWS

Configura AWS le credenziali.	Configura le impostazioni AWS CLI con cui interagisce AWS, incluso Regione AWS il formato di output che desideri utilizzare.
	`$>aws configure`
	`AWS Access Key ID`
	` [None]: <your_acc`
	`ess_key_id>`
	`AWS Secret Access Key`
	` [None]: <your_sec`
	`ret_access_key>`
	`Default region name`
	` [None]:`
	`Default output format`
	` [None]:`
	Per ulteriori informazioni, consulta la documentazione relativa ad AWS.
		Amministratore AWS

Crea un ruolo IAM.	Per creare un ruolo IAM con il ruolo di esecuzione Lambda, esegui il comando seguente.
	`aws iam create-role`
	` --role-name lambda-`
	`ex --assume-role-poli`
	`cy-document '{"Versio`
		AWS DevOps

Operazione	Description	Competenze richieste
	`n": "2012-10-17", "Statement": [{ "Effect": "Allow", "Principal": {"Service": "lambda.amazonaws.com"}, "Action": "sts:AssumeRole"}]}`	

Configura Cloud Custodian

Operazione	Description	Competenze richieste
Installa Cloud Custodian.	Per installare Cloud Custodian per il tuo sistema operativo e il tuo ambiente, segui le istruzioni nella documentazione di Cloud Custodian.	DevOps ingegnere
Controlla lo schema Cloud Custodian.	Per visualizzare l'elenco completo delle risorse Amazon RDS su cui è possibile eseguire le policy, usa il comando seguente. `custodian schema aws.rds`	DevOps ingegnere
Crea la policy Cloud Custodian.	Salva il codice contenuto nel file di policy di Cloud Custodian nella sezione Informazioni aggiuntive utilizzando un'estensione YAML.	DevOps ingegnere
Definisci le azioni di Cloud Custodian per modificare il flag accessibile al pubblico.	1. Individua il codice custode (ad esempio,). `/Users/ab`	DevOps ingegnere

Operazione	Description	Competenze richieste
	`cd/custodian/lib/python3.9/site-packages/c7n/resources/rds.py` 2. Individua la `RDSSetPublicAvailability` classe nel `rds.py` file c7n resources rds.py nella sezione Informazioni aggiuntive e modifica utilizzando il codice contenuto nel file c7n resources.	
Esegui una corsa a secco.	(Facoltativo) Per verificare quali risorse sono identificate dalla policy senza eseguire alcuna azione sulle risorse, utilizzate il comando seguente. `custodian run -dryrun <policy_name>.yaml -s <output_directory>`	DevOps ingegnere

Implementa la politica

Operazione	Description	Competenze richieste
Implementa la policy utilizzando Lambda.	Per creare la funzione Lambda che eseguirà la policy, utilizzare il comando seguente. `custodian run -s policy.yaml`	DevOps ingegnere

Operazione	Description	Competenze richieste
	Questa politica verrà quindi avviata dall' AWS CloudTrail `CreateDBInstance` evento. Di conseguenza, AWS Lambda imposterà il flag accessibile al pubblico `false` per i casi che soddisfano i criteri.	

Risorse correlate

- AWS Lambda website
- Sito Web Amazon RDS
- Documentazione Cloud Custodian

Informazioni aggiuntive

File YAML della politica di Cloud Custodian

```
policies:
  - name: "block-public-access"
    resource: rds
    description: |
      This Enforcement blocks public access for RDS instances.
    mode:
      type: cloudtrail
      events:
        - event: CreateDBInstance # Create RDS instance cloudtrail event
          source: rds.amazonaws.com
          ids: requestParameters.dBInstanceIdentifier
      role: arn:aws:iam::1234567890:role/Custodian-compliance-role
    filters:
      - type: event
        key: 'detail.requestParameters.publiclyAccessible'
        value: true
    actions:
```

```
      - type: set-public-access
        state: false
```

file rds.py di risorse c7n

```
@actions.register('set-public-access')
 class RDSSetPublicAvailability(BaseAction):

     schema = type_schema(
         "set-public-access",
         state={'type': 'boolean'})
     permissions = ('rds:ModifyDBInstance',)

     def set_accessibility(self, r):
         client = local_session(self.manager.session_factory).client('rds')
         waiter = client.get_waiter('db_instance_available')
         waiter.wait(DBInstanceIdentifier=r['DBInstanceIdentifier'])
         client.modify_db_instance(
             DBInstanceIdentifier=r['DBInstanceIdentifier'],
             PubliclyAccessible=self.data.get('state', False))


     def process(self, rds):
         with self.executor_factory(max_workers=2) as w:
             futures = {w.submit(self.set_accessibility, r): r for r in rds}
             for f in as_completed(futures):
                 if f.exception():
                     self.log.error(
                         "Exception setting public access on %s  \n %s",
                         futures[f]['DBInstanceIdentifier'], f.exception())
         return rds
```

Integrazione con Security Hub CSPM

Cloud Custodian può essere integrato con AWS Security Hub CSPMper inviare risultati di sicurezza e tentare azioni correttive. Per ulteriori informazioni, consulta Annuncio dell'integrazione di Cloud Custodian con. AWS Security Hub CSPM

Configurare l'autenticazione Windows per Amazon RDS for Microsoft SQL Server utilizzando AWS Managed Microsoft AD

Ramesh Babu Donti, Amazon Web Services

Riepilogo

Questo modello mostra come configurare l'autenticazione Windows per un Amazon Relational Database Service (Amazon RDS) per istanze AWS Directory Service for Microsoft Active Directory di SQL Server utilizzando ().AWS Managed Microsoft AD L'autenticazione Windows consente agli utenti di connettersi all'istanza RDS utilizzando le proprie credenziali di dominio anziché nomi utente e password specifici del database.

È possibile abilitare l'autenticazione di Windows durante la creazione di un nuovo database RDS SQL Server o aggiungendolo a un'istanza di database esistente. L'istanza del database si integra con AWS Managed Microsoft AD per fornire autenticazione e autorizzazione centralizzate per gli utenti del dominio che accedono al database SQL Server.

Questa configurazione migliora la sicurezza sfruttando l'infrastruttura Active Directory esistente ed elimina la necessità di gestire credenziali di database separate per gli utenti del dominio.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS con le autorizzazioni appropriate
- Un cloud privato virtuale (VPC) con quanto segue:
  - Gateway Internet e tabelle di routing configurati
  - Gateway NAT nelle sottoreti pubbliche (se per alcune istanze è richiesto l'accesso a Internet)
- AWS Identity and Access Management Ruoli (IAM):
  - Un ruolo di dominio con le seguenti politiche AWS gestite:
    - `AmazonSSMManagedInstanceCore`per abilitare AWS Systems Manager
    - `AmazonSSMDirectoryServiceAccess`per fornire le autorizzazioni per unire le istanze alle directory
  - Un ruolo di monitoraggio avanzato RDS (se il monitoraggio avanzato è abilitato)
- Gruppi di sicurezza:

  - Gruppo di sicurezza del servizio di directory per consentire le porte di comunicazione di Active Directory
  - Un gruppo di sicurezza Amazon Elastic Compute Cloud (Amazon EC2) per consentire le comunicazioni RDP 3389 e di dominio
  - Un gruppo di sicurezza RDS per consentire la 1433 portabilità di SQL Server da fonti autorizzate
- Connettività di rete:
  - Risoluzione DNS e connettività di rete corrette tra le sottoreti

Limitazioni

- Per informazioni su Regioni AWS tale supporto AWS Managed Microsoft AD con RDS per SQL Server, consulta Disponibilità delle regioni e delle versioni.
- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Architecture

Stack tecnologico di origine

- Un Active Directory locale o AWS Managed Microsoft AD

Stack tecnologico Target

- Amazon EC2
- Amazon RDS per Microsoft SQL Server
- AWS Managed Microsoft AD

Architettura Target

L'architettura include quanto segue:

- Un ruolo IAM che unisce l' EC2 istanza Amazon al AWS Managed Microsoft AD dominio.
- Un'istanza Amazon EC2 Windows per l'amministrazione e il test dei database.

- Un Amazon VPC con una sottorete privata per ospitare l'istanza Amazon RDS e le risorse interne nelle zone di disponibilità.
- Gruppi di sicurezza per il controllo degli accessi alla rete:
  - Un gruppo di sicurezza Amazon RDS per controllare l'accesso in entrata alla porta di SQL Server `1433` da fonti autorizzate.
  - Un gruppo EC2 di sicurezza Amazon per gestire l'accesso RDP tramite porte di comunicazione di porte `3389` e domini.
  - Un gruppo di sicurezza Directory Services per le comunicazioni Active Directory tramite porte `5388,389, e445`.
- AWS Managed Microsoft AD per fornire servizi centralizzati di autenticazione e autorizzazione per le risorse Windows.
- Un'istanza di database Amazon RDS for SQL Server nella sottorete privata con autenticazione Windows abilitata.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale in. Cloud AWS
- AWS Directory Serviceoffre diversi modi per utilizzare Microsoft Active Directory (AD) con altri Servizi AWS come Amazon Elastic Compute Cloud (Amazon EC2), Amazon Relational Database Service (Amazon RDS) per SQL Server e FSx Amazon per Windows File Server.
- AWS Directory Service for Microsoft Active Directoryconsente ai carichi di lavoro e alle risorse compatibili con le directory di AWS utilizzare Microsoft Active Directory in. Cloud AWS
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.

Altri servizi

- Microsoft SQL Server Management Studio (SSMS) è uno strumento per la gestione di SQL Server, che include l'accesso, la configurazione e l'amministrazione dei componenti di SQL Server.

Best practice

- Per le best practice generali, consulta Best practice for Amazon RDS.

Epiche

Configurazione AWS Managed Microsoft AD

Operazione	Description	Competenze richieste
Imposta il tipo di directory.	1. Da Console di gestione AWS, vai a AWS Directory Service. 2. Scegli Configura directory. 3. Seleziona AWS Managed Microsoft ADil tipo di directory. 4. Scegli Crea nuovo dominio AD AWS gestito, quindi scegli Avanti.	DBA, ingegnere DevOps
Configura le informazioni sulla directory.	Nella sezione Informazioni sulla directory, inserisci le informazioni richieste e mantieni i valori opzionali: 1. In Edizione, seleziona un'edizione che soddisfi i tuoi requisiti. 2. In Directory DNS name, inserisci un nome di dominio completo (FQDN). 3. In Password di amministratore, imposta una password per l'account	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
	amministratore, quindi scegli Avanti.	
Configura il VPC e le sottoreti.	1. In Rete, seleziona un VPC di destinazione (come minimo, devi configurare due sottoreti separate). AWS Zone di disponibilità 2. In Tipo di rete, seleziona Solo. IPv4 3. In Sottoreti, seleziona due sottoreti private separate AWS Zone di disponibilità, quindi scegli Avanti.	DBA DevOps , ingegnere
Rivedi e crea la directory.	1. Rivedi i valori di configura zione, quindi scegli Crea directory. 2. Attendi che lo stato della directory passi ad Attivo.	DBA, ingegnere DevOps

Creare e configurare un' EC2 istanza Amazon

Operazione	Description	Competenze richieste
Configura un'AMI per Windows.	1. Da Console di gestione AWS, vai a EC2. 2. Scegliere Launch Instance (Avvia istanza). 3. In Nome e tag, inserisci un nome e tutti i tag applicabili. 4. Scegli un'Amazon Machine Image (AMI) per Windows	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
	Server che soddisfi i tuoi requisiti. 5. In Tipo di istanza, seleziona un tipo di dimensioni adeguate. 6. In Coppia di chiavi (login), seleziona una coppia di chiavi esistente o creane una nuova.	
Configurazione delle impostazioni di rete.	1. In Impostazioni di rete, seleziona lo stesso VPC per cui viene utilizzato. AWS Directory Service 2. Scegli una sottorete privata. 3. In Firewall (gruppi di sicurezza), crea un gruppo che consenta le comunicazioni tra porte RDP 3389 e domini.	DBA, ingegnere DevOps
Configura l'archiviazione.	Configura i volumi Amazon EBS in base alle esigenze.	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
Configura i dettagli avanzati e avvia l'istanza.	1. Espandi la sezione Dettagli avanzati. 2. Per Domain join directory, seleziona quella creata in precedenza AWS Managed Microsoft AD. 3. Per il profilo dell'istanza IAM, seleziona un ruolo con le policy `AmazonSSMManagedInstanceCore` e `AmazonSSMDirectoryServiceAccess`. 4. Controlla tutti i valori di configurazione, quindi scegli Launch instance.	DBA, ingegnere DevOps

Crea e configura RDS per SQL Server con autenticazione Windows

Operazione	Description	Competenze richieste
Crea un database e configura le opzioni del motore.	1. Passa alla console Aurora e RDS e scegli Crea un database. 2. In Opzioni del motore, scegli Microsoft SQL Server. 3. In Tipo di gestione del database, selezionare Amazon RDS. 4. Per Edition, scegli un SQL Server che soddisfi i tuoi requisiti.	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
	5. Per la versione Engine, scegli l'ultima versione supportata.	
Scegliere un modello.	Scegli un modello di esempio che soddisfi le tue esigenze.	DBA, ingegnere DevOps
Configura le impostazioni del database.	1. Nella sezione Impostazioni sotto l'identificatore dell'istanza DB, inserisci un nome univoco. 2. In Nome utente principale, configura le credenziali dell'amministratore. 3. In Gestione delle credenziali, scegli Gestito in Gestione dei segreti AWS o Gestito autonomamente.	DBA, ingegnere DevOps
Configura l'istanza.	Nella sezione Configurazione dell'istanza, in Classe di istanza DB, seleziona una dimensione dell'istanza che soddisfi i tuoi requisiti.	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
Configura l'archiviazione.	1. Nella sezione Archiviazione, in Tipo di archiviazione, scegli un tipo che soddisfi i tuoi requisiti. Consigliamo gp3, io1 o io2. 2. Imposta i valori iniziali richiesti per lo storage allocato, il provisioned IOPS e il throughput di storage. 3. (Facoltativo) Espandi la sezione Configurazione dello storage aggiuntivo e seleziona Abilita scalabilità automatica dello storage.	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
Configura la connettività.	1. Nella sezione Connettività, scegli se desideri configura re una connessione a una risorsa di calcolo per il database. 2. Per VPC, scegli lo stesso VPC che hai. AWS Directory Service 3. Per il gruppo di sottoreti DB, scegli un gruppo che si estende su più zone di disponibilità. 4. Per l'accesso pubblico, scegli No. 5. Per il gruppo di sicurezza VPC (firewall), scegli un gruppo esistente o creane uno nuovo che consenta l'accesso tramite la porta SQL Server. `1433` 6. Seleziona la tua zona di disponibilità preferita. 7. Espandi la sezione Configurazione aggiuntiva e scegli se desideri utilizzar e una porta del database personalizzata.	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
Configura l'autenticazione di Windows.	1. Nella sezione Autentica zione Microsoft SQL Server Windows, selezionare la casella di controllo Abilita autenticazione Microsoft SQL Server Windows. 2. Per il tipo di autenticazione Windows, scegli AWS Managed Microsoft AD. 3. Per Directory, scegli Sfoglia directory e seleziona AWS Managed Microsoft AD.	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
Configura il monitoraggio.	1. Nella sezione Monitoraggio, scegli Standard o Advance Database Insights. 2. In Performance Insights, seleziona la casella di controllo Abilita performan ce insights. 3. Seleziona un periodo di conservazione e una AWS KMS chiave. 4. In Impostazioni di monitorag gio aggiuntive, seleziona la casella di controllo Monitoraggio avanzato. 5. (Facoltativo) In Esportazioni di log, seleziona la casella di controllo Registro degli errori. Nota: le metriche sono utili quando si desidera vedere in che modo processi o thread diversi utilizzano la CPU. Puoi anche esportare i log degli errori su Amazon CloudWatch se il log degli errori è abilitato.	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
Configura impostazioni aggiuntive.	1. Espandi la sezione Configurazione aggiuntiva. 2. Per il gruppo di parametri DB e il gruppo di opzioni DB, scegliete valori predefiniti o personalizzati. 3. Imposta il tuo fuso orario preferito. 4. Per Collation, imposta un valore. Il valore predefinito è `SQL_Latin1_General_CP1_CI_AS` . 5. In Backup: • Seleziona la casella di controllo Abilita backup automatici. In questo modo viene creata un'istantanea del database. • Per il periodo di conservazione del Backup, scegli il numero di giorni richiesto. • Per la finestra di Backup, scegli un valore. • (Facoltativo) Per la replica di Backup, scegli Abilita replica in un altro. Regione AWS • Seleziona la casella di controllo Abilita crittografia per crittografare le	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
	istanze utilizzando. AWS KMS 6. Nella finestra di manutenzione, seleziona la casella di controllo Scegli una finestra e imposta un orario preferito. 7. Seleziona la casella di controllo Abilita la protezione dall'eliminazione.	
Rivedi i costi e crea un database.	Esamina la sezione Costi mensili stimati, quindi scegli Crea database.	DBA, ingegnere DevOps

Configura l'accesso al database e verifica le connessioni

Operazione	Description	Competenze richieste
Connect al computer Windows.	Connect al computer Windows e avvia SQL Server Management Studio. 1. Usa RDP per connetterti al tuo computer Windows utilizzando AWS Managed Microsoft AD le credenziali 2. Avvia SSMS inserendo SSMS nel menu Start e seleziona SQL Server Management Studio.	DBA, ingegnere DevOps
Configura la connessione SSMS.	Configura la connessione al database utilizzando l'autenticazione di Windows.	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
	1. Quando viene visualizzata la finestra di dialogo Connetti al server (o accedendo a Esplora oggetti, Connetti, Motore di database), imposta Tipo di server su Motore di database. 2. Immettete l'endpoint RDS SQL Server (ad esempio,) `your-rds-instance.region.rds.amazonaws.com` 3. Scegli l'autenticazione Windows.	
Configurare le impostazioni di sicurezza.	Imposta i parametri di sicurezza richiesti per SSMS versione 20 o successiva. 1. Nella scheda Proprietà della connessione, imposta Crittografia su Obbligatoria. 2. Seleziona la casella di controllo Trust Server Certificate. 3. Mantieni vuoto il nome host nel campo del certificato. 4. (Facoltativo) Imposta il nome del database e regola il timeout di connessione in base alle esigenze.	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
Crea un accesso a Windows.	1. Configura e verifica l'autenticazione di Windows per gli utenti del dominio. 2. Per stabilire una connessione iniziale, scegli Connect. 3. Nella finestra di interrogazione, esegui quanto segue: `CREATE LOGIN [<domainName>\<user_name>] FROM WINDOWS; GO`	DBA, ingegnere DevOps
Prova l'autenticazione di Windows.	1. Esci dall' EC2 istanza Amazon. 2. Accedi nuovamente all' EC2 istanza utilizzando le credenziali del dominio. 3. Avvia SSMS. 4. Connect utilizzando l'autenticazione di Windows. 5. Verifica che la connessione sia riuscita.	DBA, ingegnere DevOps

Risorse correlate

- Creando il tuo AWS Managed Microsoft AD
- Unire un'istanza Amazon EC2 Windows al tuo AWS Managed Microsoft AD
- Regioni e motori DB supportati per l'autenticazione Kerberos in Amazon RDS
- Che cos'è SQL Server Management Studio (SSMS)?

Configurazione dell'accesso multi-account in Amazon DynamoDB

Shashi Dalmia, Imhortha Ojior e Esteban Serna Parra, Amazon Web Services

Riepilogo

Questo modello spiega i passaggi per configurare l'accesso tra account diversi ad Amazon DynamoDB utilizzando policy basate sulle risorse. Per i carichi di lavoro che utilizzano DynamoDB, sta diventando sempre più comune utilizzare strategie di isolamento del carico di lavoro per ridurre al minimo le minacce alla sicurezza e soddisfare i requisiti di conformità. L'implementazione di strategie di isolamento del carico di lavoro spesso richiede l'accesso tra account e regioni diverse alle risorse DynamoDB utilizzando policy basate sull'identità (IAM). AWS Identity and Access Management Ciò comporta l'impostazione delle autorizzazioni IAM e la creazione di una relazione di fiducia tra. Account AWS

Le policy basate sulle risorse per DynamoDB semplificano notevolmente il livello di sicurezza per i carichi di lavoro tra account. Questo modello fornisce passaggi e codice di esempio per dimostrare come configurare AWS Lambda le funzioni in una sola Account AWS per scrivere dati su una tabella di database DynamoDB in un account diverso.

Prerequisiti e limitazioni

Prerequisiti

- Due attivi. Account AWS Questo modello si riferisce a questi conti come Account A e Account B.
- AWS Command Line Interface (AWS CLI) installato e configurato per accedere all'Account A, per creare la tabella DynamoDB. Gli altri passaggi di questo modello forniscono istruzioni per l'utilizzo delle console IAM, DynamoDB e Lambda. Se AWS CLI invece intendi utilizzarla, configurala per accedere a entrambi gli account.

Limitazioni

- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Architecture

Il diagramma seguente mostra un'architettura a account singolo. AWS Lambda, Amazon Elastic Compute Cloud (Amazon EC2) e DynamoDB si trovano tutti nello stesso account. In questo scenario, le funzioni Lambda e le EC2 istanze Amazon possono accedere a DynamoDB. Per concedere l'accesso alla tabella DynamoDB, puoi creare una policy basata sull'identità in IAM oppure puoi creare una policy basata sulle risorse in DynamoDB.

Il diagramma seguente mostra un'architettura multi-account. Se le risorse di una tabella Account AWS richiedono l'accesso a una tabella DynamoDB in un account diverso, è necessario impostare una policy basata sulle risorse in DynamoDB per concedere l'accesso richiesto. Ad esempio, nel diagramma seguente, l'accesso alla tabella DynamoDB nell'Account A viene concesso a una funzione Lambda nell'Account B utilizzando una politica basata sulle risorse.

Questo modello descrive l'accesso tra più account tra Lambda e DynamoDB. È possibile utilizzare passaggi simili per altri Servizi AWS se le autorizzazioni appropriate sono configurate su entrambi gli account. Ad esempio, se desideri fornire a una funzione Lambda l'accesso a un bucket Amazon Simple Storage Service (Amazon S3) nell'Account A, puoi creare una policy basata sulle risorse in Amazon S3 e aggiungere le autorizzazioni al ruolo di esecuzione Lambda nell'Account B.

Tools (Strumenti)

Servizi AWS

- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.

Codice

Questo modello include un codice di esempio nella sezione Informazioni aggiuntive per mostrare come configurare una funzione Lambda nell'Account B per scrivere nella tabella DynamoDB nell'Account A. Il codice viene fornito solo a scopo illustrativo e di test. Se stai implementando questo

pattern in un ambiente di produzione, usa il codice come riferimento e personalizzalo per il tuo ambiente.

Best practice

- Segui le best practice per le policy basate sulle risorse nella documentazione di DynamoDB.
- Segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.

Epiche

Creare una policy e un ruolo IAM per la funzione Lambda nell'Account B

Operazione	Description	Competenze richieste
Crea una politica nell'Account B.	Questa policy IAM consente l'PutItemazione per una tabella DynamoDB nell'Account A. 1. Accedi all'account B in. Console di gestione AWS 2. Apri la console IAM. 3. Nel pannello di navigazione, scegli Policy e Crea policy. 4. Nella pagina Specificare le autorizzazioni, per l'editor delle politiche, seleziona JSON. 5. Immetti la seguente policy. `{ "Version": "2012-10-17", "Statement": [ { "Sid": "Statement1",`	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	<pre>"Effect": "Allow", "Action": "dynamodb:PutItem", "Resource": "arn:aws:dynamodb:<Region>:<Account-A-ID>:table/Table-Account-A" }] }</pre> 6. Sostituisci **<Region>** e **<Account-A-ID>** con i tuoi valori, quindi scegli Avanti. 7. Per Nome della politica, inserisci un nome univoco per la tua politica, ad esempio `DynamoDB-PutItem-Policy` . 8. (Facoltativo) Aggiungi una descrizione della politica. 9. Scegli Crea policy.	

Operazione	Description	Competenze richieste
Crea un ruolo nell'account B.	La funzione Lambda nell'Account B utilizza questo ruolo IAM per accedere alla tabella DynamoDB nell'Account A. 1. Apri la console IAM. 2. Nel pannello di navigazione, scegli Ruoli e quindi Crea ruolo. 3. In Select trusted entity (Seleziona entità attendibile), scegli Servizio AWS. 4. Nella sezione Caso d'uso, scegli Lambda. 5. Scegli Successivo: autorizzazioni. 6. Nella casella Politiche di filtro, inserisci DynamoDB. 7. Nell'elenco delle politiche di DynamoDB, scegli. `DynamoDB-PutItem-Policy` 8. Deseleziona la casella Politiche di filtro, quindi inserisci Lambda. 9. Nell'elenco delle politiche Lambda, scegli AWSLambda Esegui. 10. Scegli Avanti: assegna un nome, rivedi e crea. 11. Per Nome ruolo, inserisci un nome univoco per il ruolo, ad esempio	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	`DynamoDB-PutItemAccess` . 12. (Facoltativo) Aggiungi una descrizione del ruolo. 13. (Facoltativo) Aggiungi metadati al ruolo collegando i tag come coppie chiave-valore. 14. Scegli Crea ruolo. Per ulteriori informazioni sulla creazione di ruoli, consulta la documentazione IAM.	
Nota l'ARN del ruolo .	1. Aprire la console IAM. 2. Nel pannello di navigazione, selezionare Ruoli. 3. Nella casella di ricerca, inserisci `DynamoDB-PutItemAccess` , quindi scegli il ruolo. 4. Nella pagina di riepilogo del ruolo, copia l'Amazon Resource Name (ARN). Si utilizza l'ARN quando si configura la funzione Lambda.	Informazioni generali su AWS

Creare una tabella DynamoDB nell'account A

Operazione	Description	Competenze richieste

Crea una tabella DynamoDB.	Utilizzate il seguente AWS CLI comando per creare una tabella DynamoDB.
	
	`aws dynamodb create-table \`
	`--table-name Table-Account-A \`
	`--attribute-definitions \`
	`Attribute Name=category,AttributeType=S \`
	`Attribute Name=item,AttributeType=S \`
	`--key-schema \`
	`Attribute Name=category,KeyType=HASH \`
	`Attribute Name=item,KeyType=RANGE \`
	`--provisioned-throughput \`
	`ReadCapacityUnits=5,WriteCapacityUnits=5 \`
	`--resource-policy \`
	`'{`
	`"Version": "2012-10-17",`
	`"Statement": [`
	`{`
	`"Sid": "Statement1",`
		Informazioni generali su AWS

Operazione	Description	Competenze richieste
	"Effect": "Allow", "Principal": { "AWS": "arn:aws:iam::<Account-B-ID>:role/<Role-Name>" }, "Action": "dynamodb:PutItem", "Resource": "arn:aws:dynamodb:<Region>:<Account-A-ID>:table/Table-Account-A" }] }' Sostituisci quanto segue in questo esempio di codice: • `<Account-B-ID>` è l'ID dell'account B. • `<Role-Name>` è il nome del ruolo IAM che hai creato, ad esempio`DynamoDB-PutItemAccess` . • `<Region>`è il Regione AWS punto in cui si crea la tabella DynamoDB. • `<Account-A-ID>` è l'ID dell'account A.	

Operazione	Description	Competenze richieste
	Note Si specifica la configurazione della politica basata sulle risorse nell'`create-table` istruzione utilizzando il flag. `--resource-policy` Questa politica si riferisce all'ARN per la tabella DynamoDB nell'account A. Per ulteriori informazioni sulla creazione di tabelle, consulta la documentazione di DynamoDB.	

Creare una funzione Lambda nell'account B

Operazione	Description	Competenze richieste
Crea una funzione Lambda per scrivere dati su DynamoDB.	1. Accedi all'account B in. Console di gestione AWS 2. Aprire la console Lambda. 3. Nel riquadro di navigazione, scegli Funzioni, quindi seleziona Crea funzione. 4. In Nome, inserisci `lambda_write_function` .	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	5. Per Runtime, scegli Python 3.8 o versione successiva. 6. In Modifica il ruolo di esecuzione predefini to, scegli Usa un ruolo esistente. 7. Per Ruolo esistente, scegli il ruolo IAM che hai creato, ad esempio`DynamoDB-PutItemAccess` . 8. Scegli Crea funzione. 9. Nella scheda Codice, incolla il codice di esempio fornito nella sezione Informazioni aggiuntive di questo modello. Sostituis ci quanto segue in questo esempio di codice: • `<Account-A-ID>` è l'ID dell'account A. • `<Region>`è il Regione AWS luogo in cui è stata creata la tabella DynamoDB. 10. Scegliere Deploy (Implementa). 11. Scegli Test (Esegui test). Questo richiede di configura re un evento di test. Crea un nuovo evento con il tuo nome preferito, ad esempio`MyTestEve`	

Operazione	Description	Competenze richieste
	`ntForWrite` , e salva la configurazione. 12. Scegliere Test di nuovo. Viene eseguita la funzione Lambda con il nome dell'evento fornito. 13. Controllate l'output della funzione. Dovrebbe indicare che la funzione ha avuto accesso alla tabella DynamoDB nell'Account A ed è stata in grado di scrivervi dati. Per ulteriori informazioni sulla creazione di funzioni Lambda, consulta la documentazione Lambda.	

Eliminazione

Operazione	Description	Competenze richieste
Eliminare le risorse .	Per evitare di incorrere nei costi associati alle risorse create secondo questo schema, procedi come segue per eliminare tali risorse: 1. Nell'Account B, elimina la funzione Lambda che hai creato per connetterti a DynamoDB. Per istruzioni,	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	consulta la documentazione di Lambda. 2. Nell'Account A, elimina la tabella DynamoDB che hai creato. Per istruzioni, consulta la documentazione di DynamoDB. 3. Per le migliori pratiche di sicurezza, elimina la policy IAM (`DynamoDB-PutItem-Policy`) quando non è più necessaria. Per ulteriori informazioni, consulta la documentazione di IAM. 4. Per le migliori pratiche di sicurezza, elimina IAM role (`DynamoDB-PutItemAccess`) quando non è più necessario. Per ulteriori informazioni, consulta la documentazione di IAM.	

risoluzione dei problemi

Problema	Soluzione
Quando si crea la funzione Lambda, viene visualizzato un `ResourceNotFoundException` errore.	Conferma di aver inserito correttamente l'ID Regione AWS e dell'account A. Questi fanno parte dell'ARN per la tabella DynamoDB.

Risorse correlate

- Guida introduttiva a DynamoDB (documentazione su DynamoDB)
- Guida introduttiva a Lambda (documentazione Lambda)
- Utilizzo di policy basate sulle risorse per DynamoDB (documentazione DynamoDB)
- Creazione di politiche IAM (documentazione IAM)
- Logica di valutazione delle politiche tra account (documentazione IAM)
- Riferimento agli elementi della policy IAM JSON (documentazione IAM)

Informazioni aggiuntive

Codice di esempio

```
import boto3
from datetime import datetime

dynamodb_client = boto3.client('dynamodb')

def lambda_handler(event, context):
     now = datetime.now().isoformat()
     data = dynamodb_client.put_item(TableName='arn:aws:dynamodb:<Region>:<Account-
A-ID>:table/Table-Account-A', Item={"category": {"S": "Fruit"},"item": {"S":
 "Apple"},"time": {"S": now}})
     return data
```

> **Note**
>
> Quando viene creata un'istanza del client DynamoDB, viene fornito l'ARN della tabella DynamoDB anziché il nome della tabella. Ciò è necessario affinché la funzione Lambda si connetta alla tabella DynamoDB corretta durante l'esecuzione.

Configurare il routing di sola lettura in un gruppo di disponibilità Always On in SQL Server su AWS

Subhani Shaik, Amazon Web Services

Riepilogo

Questo modello illustra come utilizzare la replica secondaria in standby in SQL Server Always On trasferendo i carichi di lavoro di sola lettura dalla replica primaria alla replica secondaria.

Il mirroring del database prevede one-to-one la mappatura. Non è possibile leggere direttamente il database secondario, quindi è necessario creare istantanee. La funzionalità del gruppo di disponibilità Always On è stata introdotta in Microsoft SQL Server 2012. Nelle versioni successive, sono state introdotte funzionalità principali, incluso il routing in sola lettura. Nei gruppi di disponibilità Always On, è possibile leggere i dati direttamente dalla replica secondaria modificando la modalità di replica in sola lettura.

La soluzione Always On Availability Groups supporta l'alta disponibilità (HA), il disaster recovery (DR) e un'alternativa al mirroring del database. I gruppi di disponibilità Always On lavorano a livello di database e massimizzano la disponibilità di un set di database utente.

SQL Server utilizza il meccanismo di routing di sola lettura per reindirizzare le connessioni di sola lettura in entrata alla replica di lettura secondaria. A tale scopo, è necessario aggiungere i seguenti parametri e valori nella stringa di connessione:

- `ApplicationIntent=ReadOnly`
- `Initial Catalog=<database name>`

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo con un cloud privato virtuale (VPC), due zone di disponibilità, sottoreti private e un gruppo di sicurezza
- Due macchine Amazon Elastic Compute Cloud (Amazon EC2) con SQL Server 2019 Enterprise Edition Amazon Machine Image con Windows Server Failover Clustering (WSFC) configurate a livello di istanza e un gruppo di disponibilità Always On configurato a livello di SQL Server tra il nodo primario (`WSFCNODE1`) e il nodo secondario (`WSFCNODE2`), che fanno parte della directory AWS Directory Service per Microsoft Active Directory denominata `tagechtalk.com`

- Uno o più nodi configurati per l'accettazione nella replica secondaria `read-only`
- Un listener denominato `SQLAG1` per il gruppo di disponibilità Always On
- Motore di database SQL Server in esecuzione con lo stesso account di servizio su due nodi
- SQL Server Management Studio (SSMS)
- Un database di test denominato `test`

Versioni del prodotto

- SQL Server 2014 e versioni successive

Architecture

Stack tecnologico Target

- Amazon EC2
- AWS Managed Microsoft AD
- Amazon FSx

Architettura Target

Il diagramma seguente mostra come il listener del gruppo di disponibilità Always On (AG) reindirizza le query che contengono il `ApplicationIntent` parametro nella connessione al nodo secondario appropriato.

1. Viene inviata una richiesta al listener del gruppo di disponibilità Always On.
2. Se la stringa di connessione non contiene il `ApplicationIntent` parametro, la richiesta viene inviata all'istanza principale.
3. Se la stringa di connessione lo contiene`ApplicationIntent=ReadOnly`, la richiesta viene inviata all'istanza secondaria con configurazione di routing in sola lettura, ovvero WSFC con un gruppo di disponibilità Always On.

Tools (Strumenti)

Servizi AWS

- AWS Directory Service per Microsoft Active Directory consente ai carichi di lavoro compatibili con le directory e alle risorse AWS di utilizzare Microsoft Active Directory nel cloud AWS.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon FSx fornisce file system che supportano i protocolli di connettività standard del settore e offrono disponibilità e replica elevate in tutte le regioni AWS.

Altri servizi

- SQL Server Management Studio (SSMS) è uno strumento per la connessione, la gestione e l'amministrazione delle istanze di SQL Server.
- sqlcmd è un'utilità da riga di comando.

Best practice

Per ulteriori informazioni sui gruppi di disponibilità Always On, consulta la documentazione di SQL Server.

Epiche

Configura il routing di sola lettura

Operazione	Description	Competenze richieste
Aggiorna le repliche in modalità di sola lettura.	Per aggiornare sia la replica principale che quella secondaria in modalità di sola lettura, connettiti alla replica primaria da SSMS ed esegui il codice Step 1 dalla sezione Informazioni aggiuntive.	DBA
Crea l'URL di routing.	Per creare un URL di routing per entrambe le repliche, esegui il codice del passaggio 2 nella sezione Informazioni aggiuntive. In questo codice,	DBA

Operazione	Description	Competenze richieste
	`tagechtalk.com` è il nome della directory AWS Managed Microsoft AD.	
Crea la lista di routing.	Per creare la lista di routing per entrambe le repliche, esegui il codice del passaggio 3 nella sezione Informazioni aggiuntive.	DBA
Convalida la lista di routing.	Connect all'istanza principale da SQL Server Management Studio ed esegui il codice Step 4 dalla sezione Informazioni aggiuntive per convalidare la lista di routing.	DBA

Prova il routing di sola lettura

Operazione	Description	Competenze richieste
Connect utilizzando il ApplicationIntent parametro.	1. Da SSMS, connettiti al nome del listener del gruppo di disponibilità Always On con. `ApplicationIntent=ReadOnly;Initial Catalog=test` 2. La connessione viene stabilita con la replica secondaria. Per verificarlo, esegui il comando seguente per mostrare il nome del server connesso.	DBA

Operazione	Description	Competenze richieste
	`SELECT SERVERPROPERTY('ComputerNamePhysicalNetBios')` L'output mostrerà il nome della replica secondaria corrente (`WSFCNODE2`).	

Operazione	Description	Competenze richieste
Eseguire un failover.	1. Da SSMS, connettiti al nome del listener del gruppo di disponibilità Always On. 2. Verifica che il database primario e secondario siano sincronizzati, senza perdita di dati. 3. Esegui un failover in modo che la replica primaria corrente diventi la replica secondaria e la replica secondaria diventi la replica principale. 4. Da SSMS, connettit i al nome del listener del gruppo di disponibi lità Always On con. `ApplicationIntent= ReadOnly;Initial Catalog=test` 5. La connessione viene stabilita con la replica secondaria. Per verificarlo, mostra il nome del server connesso eseguendo il comando seguente. `SELECT SERVERPRO PERTY('Computernam ePhysicalNetBios')`	DBA

Operazione	Description	Competenze richieste
	Visualizzerà il nome corrente della replica secondaria (`WSFCNODE1`).	

Connect utilizzando l'utilità da riga di comando sqlcmd

Operazione	Description	Competenze richieste
Connect utilizzando sqlcmd.	Per connetterti da sqlcmd, esegui il codice Step 5 dalla sezione Informazioni aggiuntive del prompt dei comandi. Dopo la connessione, esegui il comando seguente per mostrare il nome del server connesso. `SELECT SERVERPROPERTY('ComputerNamePhysicalNetBios') .` L'output mostrerà il nome corrente della replica secondaria (`WSFCNODE1`).	DBA

risoluzione dei problemi

Problema	Soluzione
La creazione del listener non riesce e viene visualizzato il messaggio «Il cluster WSFC non è riuscito a portare online la risorsa Network Name».	Per informazioni, consulta il post sul blog di Microsoft Create Listener Fails with Message «Il cluster WSFC non è riuscito a portare online la risorsa Network Name».

Problema	Soluzione
Potenziali problemi, inclusi altri problemi relativi agli ascoltatori o problemi di accesso alla rete.	Vedi Risoluzione dei problemi di configurazione dei gruppi di disponibilità Always On (SQL Server) nella documentazione Microsoft.

Risorse correlate

- Configura il routing di sola lettura per un gruppo di disponibilità Always On
- Risolvi i problemi di configurazione dei gruppi di disponibilità Always On (SQL Server)

Informazioni aggiuntive

Fase 1: Aggiorna le repliche in modalità di sola lettura

```
ALTER AVAILABILITY GROUP [SQLAG1] MODIFY REPLICA ON N'WSFCNODE1' WITH (SECONDARY_ROLE
 (ALLOW_CONNECTIONS = READ_ONLY))
GO
ALTER AVAILABILITY GROUP [SQLAG1] MODIFY REPLICA ON N'WSFCNODE2' WITH (SECONDARY_ROLE
 (ALLOW_CONNECTIONS = READ_ONLY))
GO
```

Fase 2. Crea l'URL di routing

```
ALTER AVAILABILITY GROUP [SQLAG1] MODIFY REPLICA ON N'WSFCNODE1' WITH (SECONDARY_ROLE
 (READ_ONLY_ROUTING_URL = N'TCP://WSFCNode1.tagechtalk.com:1433'))
GO
ALTER AVAILABILITY GROUP [SQLAG1] MODIFY REPLICA ON N'WSFCNODE2' WITH (SECONDARY_ROLE
 (READ_ONLY_ROUTING_URL = N'TCP://WSFCNode2.tagechtalk.com:1433'))
GO
```

Fase 3. Crea la lista di routing

```
ALTER AVAILABILITY GROUP [SQLAG1] MODIFY REPLICA ON N'WSFCNODE1' WITH
 (PRIMARY_ROLE(READ_ONLY_ROUTING_LIST=('WSFCNODE2','WSFCNODE1')));
GO
ALTER AVAILABILITY GROUP [SQLAG1] MODIFY REPLICA ON N'WSFCNODE2' WITH (PRIMARY_ROLE
 (READ_ONLY_ROUTING_LIST=('WSFCNODE1','WSFCNODE2')));
```

```
GO
```

Fase 4. Convalida la lista di routing

```
SELECT AGSrc.replica_server_name AS PrimaryReplica, AGRepl.replica_server_name AS
 ReadOnlyReplica, AGRepl.read_only_routing_url AS RoutingURL , AGRL.routing_priority
 AS RoutingPriority FROM sys.availability_read_only_routing_lists AGRL INNER JOIN
 sys.availability_replicas AGSrc ON AGRL.replica_id = AGSrc.replica_id INNER JOIN
 sys.availability_replicas AGRepl ON AGRL.read_only_replica_id = AGRepl.replica_id
 INNER JOIN sys.availability_groups AV ON AV.group_id = AGSrc.group_id ORDER BY
 PrimaryReplica
```

Fase 5: Utilità di comando SQL

```
sqlcmd -S SQLAG1,1433 -E -d test -K ReadOnly
```

Connect utilizzando un tunnel SSH in pGAdmin

Jeevan Shetty e Bhanu Ganesh Gudivada, Amazon Web Services

Riepilogo

Per motivi di sicurezza, è sempre consigliabile collocare i database in una sottorete privata. Le query sul database possono essere eseguite connettendosi tramite un host bastion Amazon Elastic Compute Cloud EC2 (Amazon) in una sottorete pubblica sul cloud Amazon Web Services (AWS). Ciò richiede l'installazione di software, come pgAdmin DBeaver o, comunemente utilizzati dagli sviluppatori o dagli amministratori di database, sull'host Amazon. EC2

L'esecuzione di pGAdmin su un server Linux e l'accesso ad esso tramite un browser Web richiedono l'installazione di dipendenze aggiuntive, l'impostazione delle autorizzazioni e la configurazione.

Come soluzione alternativa, gli sviluppatori o gli amministratori di database possono connettersi a un database PostgreSQL utilizzando pgadmin per abilitare un tunnel SSH dal proprio sistema locale. In questo approccio, pgAdmin utilizza l'host EC2 Amazon nella sottorete pubblica come host intermedio prima di connettersi al database. Il diagramma nella sezione Architettura mostra la configurazione.

> **Note**
>
> Assicurati che il gruppo di sicurezza collegato al database PostgreSQL consenta la connessione sulla porta 5432 dall'host Amazon. EC2

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS esistente
- Un cloud privato virtuale (VPC) con una sottorete pubblica e una sottorete privata
- Un' EC2 istanza con un gruppo di sicurezza collegato
- Un database Edition compatibile con Amazon Aurora PostgreSQL con un gruppo di sicurezza collegato
- Una coppia di key pair Secure Shell (SSH) per configurare il tunnel

Versioni del prodotto

- pGAdmin versione 6.2+
- Amazon Aurora versione 12.7+ compatibile con PostgreSQL

Architecture

Stack tecnologico Target

- Amazon EC2
- Compatibile con Amazon Aurora PostgreSQL

Architettura Target

Il diagramma seguente mostra l'utilizzo di pGAdmin con un tunnel SSH per connettersi tramite un gateway Internet all'istanza, che si connette EC2 al database.

Tools (Strumenti)

Servizi AWS

- Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.

Altri servizi

- pgAdmin è uno strumento di gestione open source per PostgreSQL. Fornisce un'interfaccia grafica che consente di creare, gestire e utilizzare oggetti di database.

Epiche

Crea la connessione

Operazione	Description	Competenze richieste
Crea un server.	In pgAdmin, scegli Crea, quindi scegli Server. Per ulteriori informazioni sulla configurazione di pGAdmin per registrare un server, configurare una connessione e connettersi tramite tunneling SSH utilizzando la finestra di dialogo del server, vedere i collegamenti nella sezione Risorse correlate.	DBA
Fornisci un nome per il server.	Nella scheda Generale, inserisci un nome.	DBA
Inserisci i dettagli del database.	Nella scheda Connessione, inserisci i valori seguenti: • Nome/indirizzo dell'host • Porta • Database di manutenzione • Nome utente • Password	DBA
Inserisci i dettagli EC2 del server Amazon.	Nella scheda SSH Tunnel, fornisci i dettagli dell' EC2 istanza Amazon che si trova nella sottorete pubblica. • Impostare Use SSH tunneling su Sì per specifica re che pGAdmin deve	DBA

Operazione	Description	Competenze richieste
	utilizzare un tunnel SSH per la connessione al server specificato. • Nel campo Tunnel host, specificare il nome o l'indirizzo IP dell'host SSH (ad esempio, 10.x.x.x). • Nel campo Porta tunnel, specifica la porta dell'host SSH (ad esempio, 22). • Nel campo Nome utente, specifica il nome di un utente con privilegi di accesso per l'host SSH (ad esempio, ec2-user). • Specificare il tipo di autenticazione come file di identità in modo che pgAdmin utilizzi un file di chiave privata durante la connessione. • Includi la posizione del file Privacy Enhanced Mail (PEM) nel campo File di identità. Il file.pem è la coppia di EC2 chiavi Amazon.	
Salva e connettiti.	Scegli Salva per completare la configurazione e connetterti al database Aurora compatibile con PostgreSQL utilizzando il tunnel SSH.	DBA

Risorse correlate

- Dialogo del server
- Connect al server

Convertire le query JSON Oracle in SQL del database PostgreSQL

Pinesh Singal e Lokesh Gurram, Amazon Web Services

Riepilogo

Questo processo di migrazione per il passaggio dall'ambiente locale al cloud Amazon Web Services (AWS) utilizza AWS Schema Conversion Tool (AWS SCT) per convertire il codice da un database Oracle in un database PostgreSQL. La maggior parte del codice viene convertita automaticamente da AWS SCT. Tuttavia, le query Oracle relative a JSON non vengono convertite automaticamente.

A partire dalla versione Oracle 12.2, Oracle Database supporta varie funzioni JSON che aiutano a convertire i dati basati su JSON in dati basati su Row. Tuttavia, AWS SCT non converte automaticamente i dati basati su JSON in un linguaggio supportato da PostgreSQL.

Questo modello di migrazione si concentra principalmente sulla conversione manuale delle query Oracle relative a JSON con funzioni come `JSON_OBJECT` e `JSON_TABLE` da un database Oracle a un database PostgreSQL. `JSON_ARRAYAGG`

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un'istanza di database Oracle locale (attiva e funzionante)
- Un'istanza di database Amazon Relational Database Service (Amazon RDS) per PostgreSQL o Amazon Aurora PostgreSQL Compatible Edition (attiva e funzionante)

Limitazioni

- Le query `KEY` relative a JSON richiedono un formato e fisso. `VALUE` Il mancato utilizzo di quel formato restituisce un risultato errato.
- Se una modifica nella struttura JSON aggiunge nuove `KEY` `VALUE` coppie nella sezione dei risultati, è necessario modificare la procedura o la funzione corrispondente nella query SQL.
- Alcune funzioni relative a JSON sono supportate nelle versioni precedenti di Oracle e PostgreSQL ma con meno funzionalità.

Versioni del prodotto

- Oracle Database versione 12.2 e successive
- Amazon RDS for PostgreSQL o Aurora PostgreSQL versione 9.5 e successive
- Versione più recente di AWS SCT (testata utilizzando la versione 1.0.664)

Architecture

Stack tecnologico di origine

- Un'istanza di database Oracle con versione 19c

Stack tecnologico Target

- Un'istanza di database compatibile con Amazon RDS for PostgreSQL o Aurora PostgreSQL con versione 13

Architettura Target

1. Usa AWS SCT con il codice della funzione JSON per convertire il codice sorgente da Oracle a PostgreSQL.
2. La conversione produce file.sql migrati supportati da PostgreSQL.
3. Converti manualmente i codici funzione Oracle JSON non convertiti in codici funzione JSON PostgreSQL.
4. Esegui i file.sql sull'istanza DB di destinazione compatibile con Aurora PostgreSQL.

Tools (Strumenti)

Servizi AWS

- Amazon Aurora è un motore di database relazionale completamente gestito creato per il cloud e compatibile con MySQL e PostgreSQL.
- Amazon Relational Database Service (Amazon RDS) per PostgreSQL ti aiuta a configurare, gestire e scalare un database relazionale PostgreSQL nel cloud AWS.
- AWS Schema Conversion Tool (AWS SCT) supporta migrazioni di database eterogenei convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione.

Altri servizi

- Oracle SQL Developer è un ambiente di sviluppo integrato che semplifica lo sviluppo e la gestione dei database Oracle nelle implementazioni tradizionali e basate sul cloud.
- pGadmin o DBeaver. pgAdmin è uno strumento di gestione open source per PostgreSQL. Fornisce un'interfaccia grafica che consente di creare, gestire e utilizzare oggetti di database. DBeaverè uno strumento di database universale.

Best practice

La query di Oracle ha il tipo `CAST` come impostazione predefinita quando si utilizza la `JSON_TABLE` funzione. Una best practice consiste `CAST` nell'utilizzarla anche in PostgreSQL, utilizzando il doppio dei caratteri maggiori di (). >>

Per ulteriori informazioni, consulta Postgres_SQL_read_JSON nella sezione Informazioni aggiuntive.

Epiche

Genera i dati JSON nei database Oracle e PostgreSQL

Operazione	Description	Competenze richieste
Memorizza i dati JSON nel database Oracle.	Crea una tabella nel database Oracle e archivia i dati JSON nella `CLOB` colonna. Utilizzate Oracle_Table_Creation_Insert_Script che si trova nella sezione Informazioni aggiuntive.	Ingegnere della migrazione
Archivia i dati JSON nel database PostgreSQL.	Crea una tabella nel database PostgreSQL e archivia i dati JSON nella colonna. `TEXT` Usa Postgres_Table_Creation_Insert_Script che si trova nella sezione Informazioni aggiuntive.	Ingegnere della migrazione

Converti il JSON in formato ROW

Operazione	Description	Competenze richieste
Convertire i dati JSON sul database Oracle.	Scrivi una query Oracle SQL per leggere i dati JSON in formato ROW. Per ulteriori dettagli ed esempi di sintassi, vedere Oracle_SQL_read_JSON nella sezione Informazioni aggiuntive.	Ingegnere della migrazione
Converti i dati JSON nel database PostgreSQL.	Scrivi una query PostgreSQL per leggere i dati JSON in formato ROW. Per maggiori dettagli ed esempi di sintassi, consulta Postgres_SQL_read_JSON nella sezione Informazioni aggiuntive.	Ingegnere della migrazione

Converti manualmente i dati JSON utilizzando la query SQL e riporta l'output in formato JSON

Operazione	Description	Competenze richieste
Esegui aggregazioni e convalide sulla query Oracle SQL.	Per convertire manualmente i dati JSON, esegui un'unione, un'aggregazione e una convalida sulla query Oracle SQL e riporta l'output in formato JSON. Utilizza il codice in Oracle_SQL_JSON_AGGREGATION_join nella sezione Informazioni aggiuntive. 1. JOIN — I dati in formato JSON vengono passati	Ingegnere della migrazione

Operazione	Description	Competenze richieste
	come parametro di input alla query. Viene creato un JOIN interno tra questi dati statici e i dati JSON nella tabella Oracle DB. `aws_test_table` 2. Aggregazione con convalida: i dati JSON hanno `VALUE` parametri con valori come`accountNumber` , `KEY` `businessUnitId` e `parentAccountNumber` `positionId` , che vengono utilizzati per le aggregazioni. `SUM` `COUNT` 3. Formato JSON: dopo l'unione e l'aggregazione, i dati vengono riportati in formato JSON utilizzando e. `JSON_OBJECT` `JSON_ARRAYAGG`	

Operazione	Description	Competenze richieste
Esegui aggregazioni e convalide sulla query SQL di Postgres.	Per convertire manualmente i dati JSON, esegui un'unione, un'aggregazione e una convalida sulla query PostgreSQL e riporta l'output in formato JSON. Usa il codice in Postgres_SQL_JSON_Aggregation_join nella sezione Informazioni aggiuntive. 1. JOIN — I dati in formato JSON () vengono passati come parametro di input alla query della clausola. `tab1` `WITH` Viene creato un JOIN tra questi dati statici e i dati JSON, che si trovano nella tabella. `tab` Viene inoltre creato un JOIN con la `WITH` clausola, che contiene dati JSON nella tabella. `aws_test_pg_table` 2. Aggregazione: i dati JSON hanno `KEY` `VALUE` parametri con valori come`accountNumber` ,, e `parentAccountNumber businessUnitId positionId` , che vengono utilizzati per le aggregazioni e. `SUM` `COUNT` 3. Formato JSON: dopo l'unione e l'aggregazione,	Ingegnere della migrazione

Operazione	Description	Competenze richieste
	i dati vengono riportati in formato JSON utilizzando e. `JSON_BUILD_OBJECT` `JSON_AGG`	

Convertire la procedura Oracle in una funzione PostgreSQL che contiene query JSON

Operazione	Description	Competenze richieste
Convertire le query JSON nella procedura Oracle in righe.	Per la procedura Oracle di esempio, utilizzate la precedente query Oracle e il codice in ORACLE_PR OCEDURE_with_JSON_Query nella sezione Informazioni aggiuntive.	Ingegnere della migrazione
Converti le funzioni PostgreSQ L che contengono query JSON in dati basati su righe.	Per le funzioni PostgreSQL di esempio, usa la precedente query PostgreSQL e il codice che si trova in Postgres_ function_with_JSON_Query nella sezione Informazioni aggiuntive.	Ingegnere della migrazione

Risorse correlate

- Funzioni Oracle JSON
- Funzioni PostgreSQL JSON
- Esempi di funzioni JSON di Oracle
- Esempi di funzioni JSON PostgreSQL
- Strumento di conversione dello schema AWS

Informazioni aggiuntive

Per convertire il codice JSON dal database Oracle al database PostgreSQL, utilizzare i seguenti script, nell'ordine.

1. Oracle_Table_Creation_Insert_Script

```
create table aws_test_table(id number,created_on date default sysdate,modified_on
 date,json_doc clob);

REM INSERTING into EXPORT_TABLE
SET DEFINE OFF;
Insert into aws_test_table (ID,CREATED_ON,MODIFIED_ON,json_doc)
values (1,to_date('02-AUG-2022 12:30:14','DD-MON-YYYY HH24:MI:SS'),to_date('02-AUG-2022
 12:30:14','DD-MON-YYYY HH24:MI:SS'),TO_CLOB(q'[{
  "metadata" : {
    "upperLastNameFirstName" : "ABC XYZ",
    "upperEmailAddress" : "abc@gmail.com",
    "profileType" : "P"
  },
  "data" : {
    "onlineContactId" : "032323323",
    "displayName" : "Abc, Xyz",
    "firstName" : "Xyz",
    "lastName" : "Abc",
    "emailAddress" : "abc@gmail.com",
    "productRegistrationStatus" : "Not registered",
    "positionId" : "0100",
    "arrayPattern" : " -'",
    "a]')
|| TO_CLOB(q'[ccount" : {
      "companyId" : "SMGE",
      "businessUnitId" : 7,
      "accountNumber" : 42000,
      "parentAccountNumber" : 32000,
      "firstName" : "john",
      "lastName" : "doe",
      "street1" : "retOdertcaShr ",
      "city" : "new york",
      "postalcode" : "XY ABC",
      "country" : "United States"
    },
    "products" : [
```

```
      {
        "appUserGuid" : "i0acc4450000001823fbad478e2eab8a0",
        "id" : "0000000046",
]')
|| TO_CLOB(q'[        "name" : "ProView",
        "domain" : "EREADER",
        "registrationStatus" : false,
        "status" : "11"
      }
    ]
  }
}]'));
Insert into aws_test_table (ID,CREATED_ON,MODIFIED_ON,json_doc) values (2,to_date('02-
AUG-2022 12:30:14','DD-MON-YYYY HH24:MI:SS'),to_date('02-AUG-2022 12:30:14','DD-MON-
YYYY HH24:MI:SS'),TO_CLOB(q'[{
  "metadata" : {
    "upperLastNameFirstName" : "PQR XYZ",
    "upperEmailAddress" : "pqr@gmail.com",
    "profileType" : "P"
  },
  "data" : {
    "onlineContactId" : "54534343",
    "displayName" : "Xyz, pqr",
    "firstName" : "pqr",
    "lastName" : "Xyz",
    "emailAddress" : "pqr@gmail.com",
    "productRegistrationStatus" : "Not registered",
    "positionId" : "0090",
    "arrayPattern" : " -'",
    "account" : {
      "companyId" : "CARS",
      "busin]')
|| TO_CLOB(q'[essUnitId" : 6,
      "accountNumber" : 42001,
      "parentAccountNumber" : 32001,
      "firstName" : "terry",
      "lastName" : "whitlock",
      "street1" : "UO  123",
      "city" : "TOTORON",
      "region" : "NO",
      "postalcode" : "LKM 111",
      "country" : "Canada"
    },
    "products" : [
```

```
      {
        "appUserGuid" : "ia744d7790000016899f8cf3f417d6df6",
        "id" : "0000000014",
        "name" : "ProView eLooseleaf",
      ]')
|| TO_CLOB(q'[  "domain" : "EREADER",
        "registrationStatus" : false,
        "status" : "11"
      }
    ]
  }
}]'));

commit;
```

2. Postgres Table_Creation_Insert_Script

```
create table aws_test_pg_table(id int,created_on date ,modified_on date,json_doc text);
insert into aws_test_pg_table(id,created_on,modified_on,json_doc)
values(1,now(),now(),'{
  "metadata" : {
    "upperLastNameFirstName" : "ABC XYZ",
    "upperEmailAddress" : "abc@gmail.com",
    "profileType" : "P"
  },
  "data" : {
    "onlineContactId" : "032323323",
    "displayName" : "Abc, Xyz",
    "firstName" : "Xyz",
    "lastName" : "Abc",
    "emailAddress" : "abc@gmail.com",
    "productRegistrationStatus" : "Not registered",
    "positionId" : "0100",
    "arrayPattern" : " -",
    "account" : {
      "companyId" : "SMGE",
      "businessUnitId" : 7,
      "accountNumber" : 42000,
      "parentAccountNumber" : 32000,
      "firstName" : "john",
      "lastName" : "doe",
      "street1" : "retOdertcaShr ",
      "city" : "new york",
```

```
      "postalcode" : "XY ABC",
      "country" : "United States"
    },
    "products" : [
      {
        "appUserGuid" : "i0acc4450000001823fbad478e2eab8a0",
        "id" : "0000000046",
        "name" : "ProView",
        "domain" : "EREADER",
        "registrationStatus" : false,
        "status" : "11"
      }
    ]
  }
}');


insert into aws_test_pg_table(id,created_on,modified_on,json_doc)
values(2,now(),now(),'{
  "metadata" : {
    "upperLastNameFirstName" : "PQR XYZ",
    "upperEmailAddress" : "pqr@gmail.com",
    "profileType" : "P"
  },
  "data" : {
    "onlineContactId" : "54534343",
    "displayName" : "Xyz, pqr",
    "firstName" : "pqr",
    "lastName" : "Xyz",
    "emailAddress" : "a*b**@h**.k**",
    "productRegistrationStatus" : "Not registered",
    "positionId" : "0090",
    "arrayPattern" : " -",
    "account" : {
      "companyId" : "CARS",
      "businessUnitId" : 6,
      "accountNumber" : 42001,
      "parentAccountNumber" : 32001,
      "firstName" : "terry",
      "lastName" : "whitlock",
      "street1" : "UO  123",
      "city" : "TOTORON",
      "region" : "NO",
      "postalcode" : "LKM 111",
```

```
      "country" : "Canada"
    },
    "products" : [
      {
        "appUserGuid" : "ia744d7790000016899f8cf3f417d6df6",
        "id" : "0000000014",
        "name" : "ProView eLooseleaf",
        "domain" : "EREADER",
        "registrationStatus" : false,
        "status" : "11"
      }
    ]
  }
}');
```

3. Oracle_SQL_read_json

I seguenti blocchi di codice mostrano come convertire i dati Oracle JSON in formato riga.

Query e sintassi di esempio

```
SELECT   JSON_OBJECT(
 'accountCounts' VALUE JSON_ARRAYAGG(
            JSON_OBJECT(
                'businessUnitId' VALUE business_unit_id,
                        'parentAccountNumber' VALUE parent_account_number,
                        'accountNumber' VALUE account_number,
                        'totalOnlineContactsCount' VALUE online_contacts_count,
                        'countByPosition' VALUE
                    JSON_OBJECT(
                        'taxProfessionalCount' VALUE tax_count,
                        'attorneyCount' VALUE attorney_count,
                        'nonAttorneyCount' VALUE non_attorney_count,
                        'clerkCount' VALUE clerk_count
                               ) ) ) ) FROM
    (SELECT   tab_data.business_unit_id,
            tab_data.parent_account_number,
            tab_data.account_number,
            SUM(1) online_contacts_count,
            SUM(CASE WHEN tab_data.position_id = '0095' THEN  1 ELSE 0 END) tax_count,
            SUM(CASE    WHEN tab_data.position_id = '0100' THEN 1 ELSE 0 END)
 attorney_count,
            SUM(CASE    WHEN tab_data.position_id = '0090' THEN 1 ELSE 0 END)
 non_attorney_count,
```

```
           SUM(CASE    WHEN tab_data.position_id = '0050' THEN 1 ELSE 0 END)
clerk_count
       FROM aws_test_table scco,JSON_TABLE ( json_doc, '$' ERROR ON ERROR
COLUMNS (
         parent_account_number NUMBER PATH
          '$.data.account.parentAccountNumber',
           account_number NUMBER PATH '$.data.account.accountNumber',
           business_unit_id NUMBER PATH '$.data.account.businessUnitId',
           position_id VARCHAR2 ( 4 ) PATH '$.data.positionId'    )
           ) AS tab_data
           INNER JOIN JSON_TABLE ( '{
       "accounts": [{
         "accountNumber": 42000,
         "parentAccountNumber": 32000,
         "businessUnitId": 7
       }, {
         "accountNumber": 42001,
         "parentAccountNumber": 32001,
         "businessUnitId": 6
       }]
     }', '$.accounts[*]' ERROR ON ERROR
     COLUMNS (
     parent_account_number PATH '$.parentAccountNumber',
     account_number PATH '$.accountNumber',
     business_unit_id PATH '$.businessUnitId')
     ) static_data
     ON ( static_data.parent_account_number = tab_data.parent_account_number
          AND static_data.account_number = tab_data.account_number
          AND static_data.business_unit_id = tab_data.business_unit_id )
       GROUP BY
           tab_data.business_unit_id,
           tab_data.parent_account_number,
           tab_data.account_number );
```

Il documento JSON memorizza i dati come raccolte. Ogni raccolta può avere `KEY` e `VALUE` accoppiare. Ognuno `VALUE` può avere nidi `KEY` e `VALUE` coppie. La tabella seguente fornisce informazioni sulla lettura delle specifiche `VALUE` del documento JSON.

KEY	HIERARCHY o PATH da utilizzare per ottenere il VALORE	VALUE

profileType	metadata -> profileType	«P»
positionId	data -> positionId	«100"
accountNumber	data-> conto -> accountNumber	42000

Nella tabella precedente, KEY profileType è un VALUE dei metadataKEY. Il KEY positionId è uno VALUE dei dataKEY. Il KEY accountNumber è un VALUE dei accountKEY, e il account KEY è un VALUE dei dataKEY.

Esempio di documento JSON

```
{
  "metadata" : {
    "upperLastNameFirstName" : "ABC XYZ",
    "upperEmailAddress" : "abc@gmail.com",
"profileType" : "P"
  },
  "data" : {
    "onlineContactId" : "032323323",
    "displayName" : "Abc, Xyz",
    "firstName" : "Xyz",
    "lastName" : "Abc",
    "emailAddress" : "abc@gmail.com",
    "productRegistrationStatus" : "Not registered",
"positionId" : "0100",
    "arrayPattern" : " -",
    "account" : {
      "companyId" : "SMGE",
      "businessUnitId" : 7,
"accountNumber" : 42000,
      "parentAccountNumber" : 32000,
      "firstName" : "john",
      "lastName" : "doe",
      "street1" : "retOdertcaShr ",
      "city" : "new york",
      "postalcode" : "XY ABC",
      "country" : "United States"
    },
    "products" : [
```

```
      {
        "appUserGuid" : "i0acc4450000001823fbad478e2eab8a0",
        "id" : "0000000046",
        "name" : "ProView",
        "domain" : "EREADER",
        "registrationStatus" : false,
        "status" : "11"
      }
    ]
  }
}
```

Query SQL utilizzata per ottenere i campi selezionati dal documento JSON

```
select parent_account_number,account_number,business_unit_id,position_id from
 aws_test_table aws,JSON_TABLE ( json_doc, '$' ERROR ON ERROR
COLUMNS (
parent_account_number NUMBER PATH '$.data.account.parentAccountNumber',
account_number NUMBER PATH '$.data.account.accountNumber',
business_unit_id NUMBER PATH '$.data.account.businessUnitId',
position_id VARCHAR2 ( 4 ) PATH '$.data.positionId'
)) as sc
```

Nella query precedente, `JSON_TABLE` è una funzione integrata in Oracle che converte i dati JSON in formato riga. La funzione JSON_TABLE prevede parametri in formato JSON.

Ogni elemento `COLUMNS` ha un valore predefinito `PATH` e lì viene restituito un valore appropriato `VALUE` per un dato `KEY` elemento in formato riga.

Risultato della query precedente

PARENT_AC COUNT_NUMBER	NUMERO_CONTO	ID_UNITÀ_ AZIENDALE	ID_POSIZIONE
32000	42000	7	0100
32001	42001	6	0090

4. Postgres_sql_read_json

Query e sintassi di esempio

```
select *
from (
select (json_doc::json->'data'->'account'->>'parentAccountNumber')::INTEGER as
 parentAccountNumber,
(json_doc::json->'data'->'account'->>'accountNumber')::INTEGER as accountNumber,
(json_doc::json->'data'->'account'->>'businessUnitId')::INTEGER as businessUnitId,
(json_doc::json->'data'->>'positionId')::VARCHAR as positionId
from aws_test_pg_table) d ;
```

In Oracle, PATH viene utilizzato per identificare lo specifico KEY eVALUE. Tuttavia, PostgreSQL utilizza HIERARCHY un modello per la lettura e da JSON. KEY VALUE Gli stessi dati JSON menzionati di seguito vengono utilizzati negli Oracle_SQL_Read_JSON esempi seguenti.

La query SQL di tipo CAST non è consentita

(Se si forza il tipoCAST, la query ha esito negativo e viene generato un errore di sintassi).

```
select *
from (
select (json_doc::json->'data'->'account'->'parentAccountNumber') as
 parentAccountNumber,
(json_doc::json->'data'->'account'->'accountNumber')as accountNumber,
(json_doc::json->'data'->'account'->'businessUnitId') as businessUnitId,
(json_doc::json->'data'->'positionId')as positionId
from aws_test_pg_table) d ;
```

L'utilizzo di un singolo operatore maggiore di (>) restituirà il VALUE relativo valore definito. KEY Ad esempio,KEY: e:positionId. VALUE "0100"

CASTIl tipo non è consentito quando si utilizza il singolo operatore maggiore di (). >

È consentita una query SQL di tipo CAST

```
select *
from (
select (json_doc::json->'data'->'account'->>'parentAccountNumber')::INTEGER as
 parentAccountNumber,
(json_doc::json->'data'->'account'->>'accountNumber')::INTEGER as accountNumber,
(json_doc::json->'data'->'account'->>'businessUnitId')::INTEGER as businessUnitId,
(json_doc::json->'data'->>'positionId')::varchar as positionId
from aws_test_pg_table) d ;
```

Per utilizzare typeCAST, è necessario utilizzare l'operatore double greater-than. Se si utilizza il singolo operatore maggiore di, la query restituisce il valore VALUE definito (ad esempio,KEY: e:). positionId VALUE "0100" L'utilizzo dell'operatore double greater-than (>>) restituirà il valore effettivo definito per tale operazione KEY (ad esempio,KEY: eVALUE: positionId0100, senza virgolette doppie).

Nel caso precedente, is type to, parentAccountNumber is type CAST toINT, accountNumber is type CAST to INT e businessUnitId is type CAST INT to. positionId CAST VARCHAR

Le tabelle seguenti mostrano i risultati delle interrogazioni che spiegano il ruolo del singolo operatore maggiore di (>) e del doppio operatore maggiore di (). >>

Nella prima tabella, la query utilizza il singolo operatore maggiore di (). > Ogni colonna è di tipo JSON e non può essere convertita in un altro tipo di dati.

parentAccountNumber	Numero di conto	businessUnitId	ID di posizione
2003565430	2003564830	7	«0100"
2005284042	2005284042	6	«0090"
2000272719	2000272719	1	«0100"

Nella seconda tabella, la query utilizza l'operatore double greater-than (). >> Ogni colonna supporta il tipo in CAST base al valore della colonna. Ad esempio, INTEGER in questo contesto.

parentAccountNumber	Numero di conto	businessUnitId	ID di posizione
2003565430	2003564830	7	0100
2005284042	2005284042	6	0090
2000272719	2000272719	1	0100

5. Oracle_SQL_JSON_AGGREGATION_JOIN

Query di esempio

```
SELECT
    JSON_OBJECT(
        'accountCounts' VALUE JSON_ARRAYAGG(
            JSON_OBJECT(
                'businessUnitId' VALUE business_unit_id,
                        'parentAccountNumber' VALUE parent_account_number,
                        'accountNumber' VALUE account_number,
                        'totalOnlineContactsCount' VALUE online_contacts_count,
                        'countByPosition' VALUE
                    JSON_OBJECT(
                        'taxProfessionalCount' VALUE tax_count,
                        'attorneyCount' VALUE attorney_count,
                        'nonAttorneyCount' VALUE non_attorney_count,
                        'clerkCount' VALUE clerk_count
                               ) ) ) )
FROM
    (SELECT
            tab_data.business_unit_id,
            tab_data.parent_account_number,
            tab_data.account_number,
            SUM(1) online_contacts_count,
            SUM(CASE WHEN tab_data.position_id = '0095' THEN  1 ELSE 0 END) tax_count,
            SUM(CASE    WHEN tab_data.position_id = '0100' THEN 1 ELSE 0 END)
 attorney_count,
            SUM(CASE    WHEN tab_data.position_id = '0090' THEN 1 ELSE 0 END)
 non_attorney_count,
            SUM(CASE    WHEN tab_data.position_id = '0050' THEN 1 ELSE 0 END)
 clerk_count
       FROM aws_test_table scco,JSON_TABLE ( json_doc, '$' ERROR ON ERROR
 COLUMNS (
          parent_account_number NUMBER PATH
           '$.data.account.parentAccountNumber',
            account_number NUMBER PATH '$.data.account.accountNumber',
            business_unit_id NUMBER PATH '$.data.account.businessUnitId',
            position_id VARCHAR2 ( 4 ) PATH '$.data.positionId'    )
            ) AS tab_data
            INNER JOIN JSON_TABLE ( '{
       "accounts": [{
          "accountNumber": 42000,
          "parentAccountNumber": 32000,
          "businessUnitId": 7
       }, {
          "accountNumber": 42001,
```

```
          "parentAccountNumber": 32001,
          "businessUnitId": 6
        }]
      }', '$.accounts[*]' ERROR ON ERROR
      COLUMNS (
      parent_account_number PATH '$.parentAccountNumber',
      account_number PATH '$.accountNumber',
      business_unit_id PATH '$.businessUnitId')
      ) static_data
      ON ( static_data.parent_account_number = tab_data.parent_account_number
           AND static_data.account_number = tab_data.account_number
           AND static_data.business_unit_id = tab_data.business_unit_id )
        GROUP BY
            tab_data.business_unit_id,
            tab_data.parent_account_number,
            tab_data.account_number
    );
```

Per convertire i dati a livello di riga in formato JSON, Oracle dispone di funzioni integrate come`JSON_OBJECT`,`JSON_ARRAY`, `JSON_OBJECTAGG` e. `JSON_ARRAYAGG`

- `JSON_OBJECT`accetta due parametri: e. `KEY` `VALUE` Il `KEY` parametro deve essere codificato o di natura statica. Il `VALUE` parametro è derivato dall'output della tabella.
- `JSON_ARRAYAGG`accetta `JSON_OBJECT` come parametro. Questo aiuta a raggruppare l'insieme di `JSON_OBJECT` elementi in un elenco. Ad esempio, se hai un `JSON_OBJECT` elemento con più record (multipli `KEY` e `VALUE` coppie nel set di dati), `JSON_ARRAYAGG` aggiunge il set di dati e crea un elenco. Secondo il linguaggio Data Structure, `LIST` è un gruppo di elementi. In questo contesto, `LIST` è un gruppo di `JSON_OBJECT` elementi.

L'esempio seguente mostra un `JSON_OBJECT` elemento.

```
{
   "taxProfessionalCount": 0,
   "attorneyCount": 0,
   "nonAttorneyCount": 1,
   "clerkCount": 0
}
```

Il prossimo esempio mostra due `JSON_OBJECT` elementi, `LIST` indicati da parentesi quadre (`[ ]`).

```
[
    {
        "taxProfessionalCount": 0,
        "attorneyCount": 0,
        "nonAttorneyCount": 1,
        "clerkCount": 0
      }
,
    {
        "taxProfessionalCount": 2,
        "attorneyCount": 1,
        "nonAttorneyCount": 3,
        "clerkCount":4
      }
]
```

Esempio di query SQL

```
SELECT
    JSON_OBJECT(
        'accountCounts' VALUE JSON_ARRAYAGG(
            JSON_OBJECT(
                'businessUnitId' VALUE business_unit_id,
                        'parentAccountNumber' VALUE parent_account_number,
                        'accountNumber' VALUE account_number,
                        'totalOnlineContactsCount' VALUE online_contacts_count,
                        'countByPosition' VALUE
                    JSON_OBJECT(
                        'taxProfessionalCount' VALUE tax_count,
                        'attorneyCount' VALUE attorney_count,
                        'nonAttorneyCount' VALUE non_attorney_count,
                        'clerkCount' VALUE clerk_count
                               )
                        )
                                          )
              )
FROM
    (SELECT
            tab_data.business_unit_id,
            tab_data.parent_account_number,
            tab_data.account_number,
            SUM(1) online_contacts_count,
            SUM(CASE WHEN tab_data.position_id = '0095' THEN  1 ELSE   0 END
```

```
            )      tax_count,
            SUM(CASE    WHEN tab_data.position_id = '0100' THEN        1    ELSE
0 END
            )      attorney_count,

            SUM(CASE    WHEN tab_data.position_id = '0090' THEN        1    ELSE
0 END
            )      non_attorney_count,

            SUM(CASE    WHEN tab_data.position_id = '0050' THEN        1    ELSE
0 END
            )      clerk_count

        FROM
            aws_test_table scco,  JSON_TABLE ( json_doc, '$' ERROR ON ERROR
            COLUMNS (
            parent_account_number NUMBER PATH '$.data.account.parentAccountNumber',
            account_number NUMBER PATH '$.data.account.accountNumber',
            business_unit_id NUMBER PATH '$.data.account.businessUnitId',
            position_id VARCHAR2 ( 4 ) PATH '$.data.positionId'    )
            ) AS tab_data
            INNER JOIN JSON_TABLE ( '{
        "accounts": [{
          "accountNumber": 42000,
          "parentAccountNumber": 32000,
          "businessUnitId": 7
        }, {
          "accountNumber": 42001,
          "parentAccountNumber": 32001,
          "businessUnitId": 6
        }]
      }', '$.accounts[*]' ERROR ON ERROR
      COLUMNS (
      parent_account_number PATH '$.parentAccountNumber',
      account_number PATH '$.accountNumber',
      business_unit_id PATH '$.businessUnitId')
      ) static_data ON ( static_data.parent_account_number =
tab_data.parent_account_number
                         AND static_data.account_number = tab_data.account_number

                         AND static_data.business_unit_id =
tab_data.business_unit_id )
        GROUP BY
            tab_data.business_unit_id,
```

```
            tab_data.parent_account_number,
            tab_data.account_number
    );
```

Esempio di output della precedente query SQL

```
{
  "accountCounts": [
    {
      "businessUnitId": 6,
      "parentAccountNumber": 32001,
      "accountNumber": 42001,
      "totalOnlineContactsCount": 1,
      "countByPosition": {
        "taxProfessionalCount": 0,
        "attorneyCount": 0,
        "nonAttorneyCount": 1,
        "clerkCount": 0
      }
    },
    {
      "businessUnitId": 7,
      "parentAccountNumber": 32000,
      "accountNumber": 42000,
      "totalOnlineContactsCount": 1,
      "countByPosition": {
        "taxProfessionalCount": 0,
        "attorneyCount": 1,
        "nonAttorneyCount": 0,
        "clerkCount": 0
      }
    }
  ]
}
```

6. Postgres_SQL_JSON_AGGREGATION_JOIN

Le `JSON_BUILD_OBJECT` funzioni `JSON_AGG` integrate di PostgreSQL convertono i dati a livello di riga in formato JSON. `JSON_BUILD_OBJECT`PostgreSQL e sono equivalenti a Oracle `JSON_AGG` e. `JSON_OBJECT` `JSON_ARRAYAGG`

Query di esempio

```
select
JSON_BUILD_OBJECT ('accountCounts',
    JSON_AGG(
        JSON_BUILD_OBJECT ('businessUnitId',businessUnitId
        ,'parentAccountNumber',parentAccountNumber
        ,'accountNumber',accountNumber
        ,'totalOnlineContactsCount',online_contacts_count,
        'countByPosition',
            JSON_BUILD_OBJECT (
            'taxProfessionalCount',tax_professional_count
            ,'attorneyCount',attorney_count
            ,'nonAttorneyCount',non_attorney_count
            ,'clerkCount',clerk_count
            )
        )
    )
)
from (
with tab as (select * from (
select (json_doc::json->'data'->'account'->>'parentAccountNumber')::INTEGER as
 parentAccountNumber,
(json_doc::json->'data'->'account'->>'accountNumber')::INTEGER as accountNumber,
(json_doc::json->'data'->'account'->>'businessUnitId')::INTEGER as businessUnitId,
(json_doc::json->'data'->>'positionId')::varchar as positionId
from aws_test_pg_table) a ) ,
tab1 as ( select
(json_array_elements(b.jc -> 'accounts') ->> 'accountNumber')::integer accountNumber,
(json_array_elements(b.jc -> 'accounts') ->> 'businessUnitId')::integer
 businessUnitId,
(json_array_elements(b.jc -> 'accounts') ->> 'parentAccountNumber')::integer
 parentAccountNumber
from (
select '{
        "accounts": [{
          "accountNumber": 42001,
          "parentAccountNumber": 32001,
          "businessUnitId": 6
        }, {
          "accountNumber": 42000,
          "parentAccountNumber": 32000,
          "businessUnitId": 7
        }]
      }'::json as jc) b)
```

```
select
tab.businessUnitId::text,
tab.parentAccountNumber::text,
tab.accountNumber::text,
SUM(1) online_contacts_count,
SUM(CASE WHEN tab.positionId::text = '0095' THEN 1 ELSE 0  END)
 tax_professional_count,
SUM(CASE WHEN tab.positionId::text = '0100' THEN 1 ELSE 0  END)      attorney_count,
SUM(CASE  WHEN tab.positionId::text = '0090' THEN      1  ELSE      0 END)
 non_attorney_count,
SUM(CASE  WHEN tab.positionId::text = '0050' THEN      1  ELSE      0 END)
 clerk_count
from tab1,tab
where tab.parentAccountNumber::INTEGER=tab1.parentAccountNumber::INTEGER
and tab.accountNumber::INTEGER=tab1.accountNumber::INTEGER
and tab.businessUnitId::INTEGER=tab1.businessUnitId::INTEGER
GROUP BY      tab.businessUnitId::text,
            tab.parentAccountNumber::text,
            tab.accountNumber::text) a;
```

Esempio di output della query precedente

L'output di Oracle e PostgreSQL è esattamente lo stesso.

```
{
  "accountCounts": [
    {
      "businessUnitId": 6,
      "parentAccountNumber": 32001,
      "accountNumber": 42001,
      "totalOnlineContactsCount": 1,
      "countByPosition": {
        "taxProfessionalCount": 0,
        "attorneyCount": 0,
        "nonAttorneyCount": 1,
        "clerkCount": 0
      }
    },
    {
      "businessUnitId": 7,
      "parentAccountNumber": 32000,
      "accountNumber": 42000,
      "totalOnlineContactsCount": 1,
      "countByPosition": {
```

```
        "taxProfessionalCount": 0,
        "attorneyCount": 1,
        "nonAttorneyCount": 0,
        "clerkCount": 0
      }
    }
  ]
}
```

7. Oracle_procedure_with_JSON_Query

Questo codice converte la procedura Oracle in una funzione PostgreSQL con query SQL JSON. Mostra come la query traspone JSON in righe e viceversa.

```
CREATE OR REPLACE PROCEDURE p_json_test(p_in_accounts_json IN varchar2,
 p_out_accunts_json  OUT varchar2)
IS
BEGIN
/*
p_in_accounts_json paramter should have following format:
       {
        "accounts": [{
          "accountNumber": 42000,
          "parentAccountNumber": 32000,
          "businessUnitId": 7
        }, {
          "accountNumber": 42001,
          "parentAccountNumber": 32001,
          "businessUnitId": 6
        }]
      }
*/
SELECT
    JSON_OBJECT(
        'accountCounts' VALUE JSON_ARRAYAGG(
            JSON_OBJECT(
                'businessUnitId' VALUE business_unit_id,
                        'parentAccountNumber' VALUE parent_account_number,
                        'accountNumber' VALUE account_number,
                        'totalOnlineContactsCount' VALUE online_contacts_count,
                        'countByPosition' VALUE
                    JSON_OBJECT(
                        'taxProfessionalCount' VALUE tax_count,
```

```
                        'attorneyCount' VALUE attorney_count,
                        'nonAttorneyCount' VALUE non_attorney_count,
                        'clerkCount' VALUE clerk_count
                               ) ) ) )
into p_out_accunts_json
FROM
    (SELECT
            tab_data.business_unit_id,
            tab_data.parent_account_number,
            tab_data.account_number,
            SUM(1) online_contacts_count,
            SUM(CASE WHEN tab_data.position_id = '0095' THEN  1 ELSE 0 END) tax_count,
            SUM(CASE    WHEN tab_data.position_id = '0100' THEN 1 ELSE 0 END)
 attorney_count,
            SUM(CASE    WHEN tab_data.position_id = '0090' THEN 1 ELSE 0 END)
 non_attorney_count,
            SUM(CASE    WHEN tab_data.position_id = '0050' THEN 1 ELSE 0 END)
 clerk_count
        FROM aws_test_table scco,JSON_TABLE ( json_doc, '$' ERROR ON ERROR
            COLUMNS (
            parent_account_number NUMBER PATH '$.data.account.parentAccountNumber',
            account_number NUMBER PATH '$.data.account.accountNumber',
            business_unit_id NUMBER PATH '$.data.account.businessUnitId',
            position_id VARCHAR2 ( 4 ) PATH '$.data.positionId'    )
            ) AS tab_data
            INNER JOIN JSON_TABLE ( p_in_accounts_json, '$.accounts[*]' ERROR ON ERROR

      COLUMNS (
      parent_account_number PATH '$.parentAccountNumber',
      account_number PATH '$.accountNumber',
      business_unit_id PATH '$.businessUnitId')
      ) static_data
      ON ( static_data.parent_account_number = tab_data.parent_account_number
           AND static_data.account_number = tab_data.account_number
           AND static_data.business_unit_id = tab_data.business_unit_id )
        GROUP BY
            tab_data.business_unit_id,
            tab_data.parent_account_number,
            tab_data.account_number
    );
EXCEPTION
WHEN OTHERS THEN
   raise_application_error(-20001,'Error while running the JSON query');
END;
```

```
/
```

Esecuzione della procedura

Il seguente blocco di codice spiega come eseguire la procedura Oracle creata in precedenza con un esempio di input JSON per la procedura. Fornisce inoltre il risultato o l'output di questa procedura.

```
set serveroutput on;
declare
v_out varchar2(30000);
v_in varchar2(30000):= '{
        "accounts": [{
          "accountNumber": 42000,
          "parentAccountNumber": 32000,
          "businessUnitId": 7
        }, {
          "accountNumber": 42001,
          "parentAccountNumber": 32001,
          "businessUnitId": 6
        }]
      }';
begin
  p_json_test(v_in,v_out);
  dbms_output.put_line(v_out);
end;
/
```

Output della procedura

```
{
  "accountCounts": [
    {
      "businessUnitId": 6,
      "parentAccountNumber": 32001,
      "accountNumber": 42001,
      "totalOnlineContactsCount": 1,
      "countByPosition": {
        "taxProfessionalCount": 0,
        "attorneyCount": 0,
        "nonAttorneyCount": 1,
        "clerkCount": 0
      }
    },
```

```
    {
      "businessUnitId": 7,
      "parentAccountNumber": 32000,
      "accountNumber": 42000,
      "totalOnlineContactsCount": 1,
      "countByPosition": {
        "taxProfessionalCount": 0,
        "attorneyCount": 1,
        "nonAttorneyCount": 0,
        "clerkCount": 0
      }
    }
  ]
}
```

8. Postgres_function_with_JSON_QUERY

Funzione di esempio

```
CREATE OR REPLACE  FUNCTION f_pg_json_test(p_in_accounts_json  text)
RETURNS text
LANGUAGE plpgsql
AS
$$
DECLARE
 v_out_accunts_json   text;
BEGIN
SELECT
JSON_BUILD_OBJECT ('accountCounts',
    JSON_AGG(
        JSON_BUILD_OBJECT ('businessUnitId',businessUnitId
        ,'parentAccountNumber',parentAccountNumber
        ,'accountNumber',accountNumber
        ,'totalOnlineContactsCount',online_contacts_count,
        'countByPosition',
            JSON_BUILD_OBJECT (
            'taxProfessionalCount',tax_professional_count
            ,'attorneyCount',attorney_count
            ,'nonAttorneyCount',non_attorney_count
            ,'clerkCount',clerk_count
            ))))
INTO v_out_accunts_json
FROM (
WITH tab AS (SELECT * FROM (
```

```
SELECT (json_doc::json->'data'->'account'->>'parentAccountNumber')::INTEGER AS
 parentAccountNumber,
(json_doc::json->'data'->'account'->>'accountNumber')::INTEGER AS accountNumber,
(json_doc::json->'data'->'account'->>'businessUnitId')::INTEGER AS businessUnitId,
(json_doc::json->'data'->>'positionId')::varchar AS positionId
FROM aws_test_pg_table) a ) ,
tab1 AS ( SELECT
(json_array_elements(b.jc -> 'accounts') ->> 'accountNumber')::integer accountNumber,
(json_array_elements(b.jc -> 'accounts') ->> 'businessUnitId')::integer businessUnitId,
(json_array_elements(b.jc -> 'accounts') ->> 'parentAccountNumber')::integer
 parentAccountNumber
FROM (
SELECT p_in_accounts_json::json AS jc) b)
SELECT
tab.businessUnitId::text,
tab.parentAccountNumber::text,
tab.accountNumber::text,
SUM(1) online_contacts_count,
SUM(CASE WHEN tab.positionId::text = '0095' THEN 1 ELSE 0  END)
 tax_professional_count,
SUM(CASE WHEN tab.positionId::text = '0100' THEN 1 ELSE 0  END)      attorney_count,
SUM(CASE  WHEN tab.positionId::text = '0090' THEN      1  ELSE      0 END)
 non_attorney_count,
SUM(CASE  WHEN tab.positionId::text = '0050' THEN      1  ELSE      0 END)
 clerk_count
FROM tab1,tab
WHERE tab.parentAccountNumber::INTEGER=tab1.parentAccountNumber::INTEGER
AND tab.accountNumber::INTEGER=tab1.accountNumber::INTEGER
AND tab.businessUnitId::INTEGER=tab1.businessUnitId::INTEGER
GROUP BY      tab.businessUnitId::text,
            tab.parentAccountNumber::text,
            tab.accountNumber::text) a;
RETURN v_out_accunts_json;
END;
$$;
```

Esecuzione della funzione

```
select    f_pg_json_test('{
        "accounts": [{
          "accountNumber": 42001,
          "parentAccountNumber": 32001,
          "businessUnitId": 6
```

```
        }, {
          "accountNumber": 42000,
          "parentAccountNumber": 32000,
          "businessUnitId": 7
        }]
      }')   ;
```

Uscita della funzione

L'output seguente è simile all'output della procedura Oracle. La differenza è che questo output è in formato testo.

```
{
  "accountCounts": [
    {
      "businessUnitId": "6",
      "parentAccountNumber": "32001",
      "accountNumber": "42001",
      "totalOnlineContactsCount": 1,
      "countByPosition": {
        "taxProfessionalCount": 0,
        "attorneyCount": 0,
        "nonAttorneyCount": 1,
        "clerkCount": 0
      }
    },
    {
      "businessUnitId": "7",
      "parentAccountNumber": "32000",
      "accountNumber": "42000",
      "totalOnlineContactsCount": 1,
      "countByPosition": {
        "taxProfessionalCount": 0,
        "attorneyCount": 1,
        "nonAttorneyCount": 0,
        "clerkCount": 0
      }
    }
  ]
}
```

Copia le tabelle Amazon DynamoDB tra gli account utilizzando AWS Backup

Ramkumar Ramanujam, Amazon Web Services

Riepilogo

Quando si lavora con Amazon DynamoDB AWS on, un caso d'uso comune consiste nel copiare o sincronizzare le tabelle DynamoDB in ambienti di sviluppo, test o staging con i dati delle tabelle presenti nell'ambiente di produzione. Come prassi standard, ogni ambiente ne utilizza uno diverso. Account AWS

AWS Backup supporta il backup e il ripristino dei dati tra regioni e account diversi per DynamoDB, Amazon Simple Storage Service (Amazon S3) e altri. Servizi AWS Questo modello fornisce i passaggi per utilizzare il backup e il ripristino AWS Backup tra account diversi per copiare le tabelle DynamoDB tra di loro. Account AWS

Prerequisiti e limitazioni

Prerequisiti

- Due attivi Account AWS che appartengono alla stessa organizzazione in AWS Organizations
- Autorizzazioni per creare tabelle DynamoDB in entrambi gli account
- AWS Identity and Access Management (IAM) autorizzazioni per creare e utilizzare casseforti AWS Backup

Limitazioni

- L'origine e la destinazione Account AWS devono far parte della stessa organizzazione in. AWS Organizations

Architecture

Stack tecnologico di destinazione

- AWS Backup
- Amazon DynamoDB

Architettura di destinazione

1. Crea il backup della tabella DynamoDB nell'archivio di backup AWS Backup nell'account di origine.
2. Copia il backup nell'archivio di backup nell'account di destinazione.
3. Ripristina la tabella DynamoDB nell'account di destinazione utilizzando il backup dall'archivio di backup nell'account di destinazione.

Automazione e scalabilità

È possibile AWS Backup utilizzarlo per pianificare l'esecuzione dei backup a intervalli specifici.

Tools (Strumenti)

- AWS Backupè un servizio completamente gestito per centralizzare e automatizzare la protezione dei dati su Servizi AWS, nel cloud e in locale. Utilizzando questo servizio, è possibile configurare le politiche di backup e monitorare l'attività delle risorse in un unico posto. AWS Consente di automatizzare e consolidare le attività di backup precedentemente eseguite servizio per servizio ed elimina la necessità di creare script personalizzati e processi manuali.
- Amazon DynamoDB è un servizio di database NoSQL completamente gestito che offre prestazioni veloci e prevedibili con una scalabilità perfetta.

Epiche

Attiva le AWS Backup funzionalità negli account di origine e di destinazione

Operazione	Description	Competenze richieste
Attiva le funzionalità avanzate per DynamoDB e il backup tra account.	Sia nell'origine che nella destinazione Account AWS, procedi come segue: 1. Sul Console di gestione AWS, apri la AWS Backup console. 2. Seleziona Impostazioni.	AWS DevOps, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	3. In Funzionalità avanzate per i backup di Amazon DynamoDB, verifica che le funzionalità avanzate siano abilitate o scegli Abilita. 4. In Gestione tra account, per il backup su più account, scegli Abilita.	

Crea archivi di backup negli account di origine e di destinazione

Operazione	Description	Competenze richieste
Crea casseforti di backup.	Sia nell'origine che nella destinazione Account AWS, procedi come segue: 1. Sulla AWS Backup console, scegli Backup vault. 2. Scegliere Crea vault di Backup. 3. Copia l'Amazon Resource Name (ARN) del backup vault e salvalo. L' ARNs archivio di backup di origine e quello di destinazione saranno necessari quando si copia il backup della tabella DynamoDB tra gli account di origine e di destinazione.	AWS DevOps, ingegnere addetto alla migrazione

Esegui backup e ripristino utilizzando gli archivi di backup

Operazione	Description	Competenze richieste
Nell'account di origine, crea un backup della tabella DynamoDB.	Per creare un backup per la tabella DynamoDB nell'account di origine, procedi come segue: 1. Nella pagina AWS Backup Dashboard, scegli Crea backup su richiesta. 2. Nella sezione Impostazioni, per Tipo di risorsa, seleziona DynamoDB, quindi seleziona il nome della tabella. 3. Nell'elenco a discesa Backup vault, seleziona l'archivio di backup che hai creato nell'account di origine. 4. Seleziona il periodo di conservazione che desideri. 5. Seleziona Crea backup on demand. Viene creato un nuovo processo di backup. Per monitorare lo stato del processo di backup, nella pagina AWS Backup Processi, selezionare la scheda Processi di backup. Tutti i processi di backup attivi,	AWS DevOps, DBA, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	in corso e completati sono elencati in questa scheda.	

Operazione	Description	Competenze richieste
Copia il backup dall'account di origine all'account di destinazione.	Una volta completato il processo di backup, copia il backup della tabella DynamoDB dall'archivio di backup nell'account di origine all'archivio di backup nell'account di destinazione. Per copiare il backup vault, nell'account di origine, procedi come segue: 1. Sulla AWS Backup console, scegli Backup vault. 2. In Backup, seleziona il backup della tabella DynamoDB. 3. Selezionare Actions (Operazioni), Copy (Copia). 4. Inserisci l'account Regione AWS di destinazione. 5. Per ARN del vault esterno, inserisci l'ARN dell'archivio di backup che hai creato nell'account di destinazione. 6. Per copiare i backup dall'account di origine all'account di destinazione, nell'archivio di backup dell'account di destinazione, abilita l'accesso da un account diverso.	AWS DevOps, ingegnere addetto alla migrazione, DBA

Operazione	Description	Competenze richieste
Ripristina il backup nell'account di destinazione.	Nella destinazione Account AWS, procedi come segue: 1. Sulla AWS Backup console, scegli Backup vault. 2. In Backup, seleziona il backup che hai copiato dall'account di origine. 3. Scegli Azioni, Ripristina. 4. Immettete il nome della tabella DynamoDB di destinazione che desiderate ripristinare.	AWS DevOps, DBA, ingegnere addetto alla migrazione

Risorse correlate

- Utilizzo AWS Backup con DynamoDB
- Creazione di copie di backup su Account AWS
- AWS Backup prezzi

Copia le tabelle Amazon DynamoDB tra gli account utilizzando un'implementazione personalizzata

Ramkumar Ramanujam, Amazon Web Services

Riepilogo

Quando si lavora con Amazon DynamoDB su Amazon Web Services (AWS), un caso d'uso comune consiste nel copiare o sincronizzare le tabelle DynamoDB in ambienti di sviluppo, test o staging con i dati delle tabelle presenti nell'ambiente di produzione. Come prassi standard, ogni ambiente utilizza un account AWS diverso.

DynamoDB ora supporta il backup su più account utilizzando AWS Backup. Per informazioni sui costi di storage associati all'utilizzo di AWS Backup, consulta i prezzi di AWS Backup. Quando usi AWS Backup per copiare più account, gli account di origine e di destinazione devono far parte di un'organizzazione AWS Organizations. Esistono altre soluzioni per il backup e il ripristino tra account che utilizzano servizi AWS come AWS Glue. L'utilizzo di queste soluzioni, tuttavia, aumenta l'ingombro delle applicazioni, poiché ci sono più servizi AWS da distribuire e mantenere.

Puoi anche usare Amazon DynamoDB Streams per acquisire le modifiche alle tabelle nell'account di origine. Quindi puoi avviare una funzione AWS Lambda e apportare le modifiche corrispondenti nella tabella di destinazione nell'account di destinazione. Ma questa soluzione si applica ai casi d'uso in cui le tabelle di origine e di destinazione devono essere sempre mantenute sincronizzate. Potrebbe non essere applicabile agli ambienti di sviluppo, test e gestione temporanea in cui i dati vengono aggiornati frequentemente.

Questo modello fornisce i passaggi per implementare una soluzione personalizzata per copiare una tabella Amazon DynamoDB da un account all'altro. Questo modello può essere implementato utilizzando linguaggi di programmazione comuni come C#, Java e Python. Ti consigliamo di utilizzare un linguaggio supportato da un SDK AWS.

Prerequisiti e limitazioni

Prerequisiti

- Due account AWS attivi
- Tabelle DynamoDB in entrambi gli account
- Conoscenza dei ruoli e delle policy di AWS Identity and Access Management (IAM)

- Conoscenza di come accedere alle tabelle di Amazon DynamoDB utilizzando qualsiasi linguaggio di programmazione comune, come C#, Java o Python

Limitazioni

Questo modello si applica alle tabelle DynamoDB di dimensioni pari o inferiori a 2 GB. Con una logica aggiuntiva per gestire le interruzioni della connessione o della sessione, le limitazioni, gli errori e i nuovi tentativi, può essere utilizzato per tabelle più grandi.

L'operazione di scansione DynamoDB, che legge gli elementi dalla tabella di origine, può recuperare solo fino a 1 MB di dati in una singola chiamata. Per le tabelle più grandi, superiori a 2 GB, questa limitazione può aumentare il tempo totale necessario per eseguire una copia completa della tabella.

Architecture

Il diagramma seguente mostra l'implementazione personalizzata tra gli account AWS di origine e di destinazione. Le politiche e i token di sicurezza IAM vengono utilizzati con l'implementazione personalizzata. I dati vengono letti da Amazon DynamoDB nell'account di origine e scritti su DynamoDB nell'account di destinazione.

Automazione e scalabilità

Questo modello si applica alle tabelle DynamoDB di dimensioni inferiori, circa 2 GB.

Per applicare questo modello a tabelle più grandi, risolvi i seguenti problemi:

- Durante l'operazione di copia della tabella, vengono mantenute due sessioni attive, utilizzando token di sicurezza diversi. Se l'operazione di copia della tabella richiede più tempo dei tempi di scadenza del token, è necessario implementare una logica per aggiornare i token di sicurezza.
- Se non vengono fornite unità di capacità di lettura (RCUs) e unità di capacità di scrittura (WCUs) sufficienti, le operazioni di lettura o scrittura sulla tabella di origine o di destinazione potrebbero risultare limitate. Assicurati di catturare e gestire queste eccezioni.
- Gestite eventuali altri errori o eccezioni e predisponete un meccanismo di ripetizione dei tentativi per riprovare o continuare dal punto in cui l'operazione di copia non è riuscita.

Tools (Strumenti)

Strumenti

- Amazon DynamoDB — Amazon DynamoDB è un servizio di database NoSQL completamente gestito che offre prestazioni veloci e prevedibili con una scalabilità perfetta.
- Gli strumenti aggiuntivi richiesti differiranno in base al linguaggio di programmazione scelto per l'implementazione. Ad esempio, se usi C#, avrai bisogno di Microsoft Visual Studio e dei seguenti NuGet pacchetti:
  - `AWSSDK`
  - `AWSSDK.DynamoDBv2`

Codice

Il seguente frammento di codice Python elimina e ricrea una tabella DynamoDB utilizzando la libreria Boto3.

Non utilizzare la `AWS_ACCESS_KEY_ID` mano `AWS_SECRET_ACCESS_KEY` di un utente IAM perché si tratta di credenziali a lungo termine, che dovrebbero essere evitate per l'accesso programmatico ai servizi AWS. Per ulteriori informazioni sulle credenziali temporanee, consulta la sezione Best practice.

Le `AWS_ACCESS_KEY_IDAWS_SECRET_ACCESS_KEY`, e `TEMPORARY_SESSION_TOKEN` utilizzate nel seguente frammento di codice sono credenziali temporanee recuperate da AWS Security Token Service (AWS STS).

```
import boto3
import sys
import json

#args = input-parameters = GLOBAL_SEC_INDEXES_JSON_COLLECTION,
 ATTRIBUTES_JSON_COLLECTION, TARGET_DYNAMODB_NAME, TARGET_REGION, ...

#Input param: GLOBAL_SEC_INDEXES_JSON_COLLECTION
#[{"IndexName":"Test-index","KeySchema":[{"AttributeName":"AppId","KeyType":"HASH"},
{"AttributeName":"AppType","KeyType":"RANGE"}],"Projection":
{"ProjectionType":"INCLUDE","NonKeyAttributes":["PK","SK","OwnerName","AppVersion"]}}]

#Input param: ATTRIBUTES_JSON_COLLECTION
#[{"AttributeName":"PK","AttributeType":"S"},
{"AttributeName":"SK","AttributeType":"S"},
{"AttributeName":"AppId","AttributeType":"S"},
{"AttributeName":"AppType","AttributeType":"N"}]
```

```
region = args['TARGET_REGION']
target_ddb_name = args['TARGET_DYNAMODB_NAME']

global_secondary_indexes = json.loads(args['GLOBAL_SEC_INDEXES_JSON_COLLECTION'])
attribute_definitions = json.loads(args['ATTRIBUTES_JSON_COLLECTION'])

# Drop and create target DynamoDB table
dynamodb_client = boto3.Session(
        aws_access_key_id=args['AWS_ACCESS_KEY_ID'],
        aws_secret_access_key=args['AWS_SECRET_ACCESS_KEY'],
        aws_session_token=args['TEMPORARY_SESSION_TOKEN'],
    ).client('dynamodb')

# Delete table
print('Deleting table: ' + target_ddb_name + ' ...')

try:
    dynamodb_client.delete_table(TableName=target_ddb_name)

    #Wait for table deletion to complete
    waiter = dynamodb_client.get_waiter('table_not_exists')
    waiter.wait(TableName=target_ddb_name)
    print('Table deleted.')
except dynamodb_client.exceptions.ResourceNotFoundException:
    print('Table already deleted / does not exist.')
    pass

print('Creating table: ' + target_ddb_name + ' ...')

table = dynamodb_client.create_table(
    TableName=target_ddb_name,
    KeySchema=[
        {
            'AttributeName': 'PK',
            'KeyType': 'HASH'  # Partition key
        },
        {
            'AttributeName': 'SK',
            'KeyType': 'RANGE'  # Sort key
        }
    ],
    AttributeDefinitions=attribute_definitions,
    GlobalSecondaryIndexes=global_secondary_indexes,
    BillingMode='PAY_PER_REQUEST'
```

```
)

waiter = dynamodb_client.get_waiter('table_exists')
waiter.wait(TableName=target_ddb_name)

print('Table created.')
```

Best practice

Credenziali temporanee

Come best practice di sicurezza, mentre accedi ai servizi AWS in modo programmatico, evita di utilizzare la `AWS_ACCESS_KEY_ID` e `AWS_SECRET_ACCESS_KEY` di un utente IAM perché si tratta di credenziali a lungo termine. Cerca sempre di utilizzare credenziali temporanee per accedere ai servizi AWS in modo programmatico.

Ad esempio, uno sviluppatore inserisce nell'applicazione il codice fisso `AWS_ACCESS_KEY_ID` e `AWS_SECRET_ACCESS_KEY` di un utente IAM durante lo sviluppo, ma non riesce a rimuovere i valori codificati prima di apportare le modifiche al code repository. Queste credenziali esposte possono essere utilizzate da utenti indesiderati o malintenzionati, il che può avere gravi implicazioni (specialmente se le credenziali esposte hanno privilegi di amministratore). Queste credenziali esposte devono essere disattivate o eliminate immediatamente utilizzando la console IAM o AWS Command Line Interface (AWS CLI).

Per ottenere credenziali temporanee per l'accesso programmatico ai servizi AWS, usa AWS STS. Le credenziali temporanee sono valide solo per il tempo specificato (da 15 minuti a 36 ore). La durata massima consentita per le credenziali temporanee varia in base a fattori quali le impostazioni dei ruoli e il concatenamento dei ruoli. Per ulteriori informazioni su AWS STS, consulta la documentazione.

Poemi epici

Configurare le tabelle DynamoDB

Operazione	Description	Competenze richieste
Crea tabelle DynamoDB.	Crea tabelle DynamoDB, con indici, negli account AWS di origine e di destinazione.	Sviluppatore di app, DBA, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	Imposta il provisioning della capacità come modalità on-demand, che consente a DynamoDB di scalare le read/write capacità in modo dinamico in base al carico di lavoro. In alternativa, è possibile utilizzare la capacità assegnata con 4000 e 4000. RCUs WCUs	
Compila la tabella dei sorgenti.	Compila la tabella DynamoDB nell'account di origine con i dati di test. Avere almeno 50 MB o più di dati di test consente di visualizzare il picco e la media del RCUs consumo durante la copia della tabella. È quindi possibile modificare il provisioning della capacità in base alle esigenze.	Sviluppatore di app, DBA, ingegnere addetto alla migrazione

Imposta le credenziali per accedere alle tabelle DynamoDB

Operazione	Description	Competenze richieste
Crea ruoli IAM per accedere alle tabelle DynamoDB di origine e di destinazione.	Crea un ruolo IAM nell'account di origine con le autorizzazioni per accedere (leggere) alla tabella DynamoDB nell'acco unt di origine.	Sviluppatore di app, AWS DevOps

Operazione	Description	Competenze richieste
	Aggiungi l'account di origine come entità attendibile per questo ruolo. Crea un ruolo IAM nell'account di destinazione con le autorizzazioni per accedere (creare, leggere, aggiornare, eliminare) alla tabella DynamoDB nell'account di destinazione. Aggiungi l'account di destinazione come entità affidabile per questo ruolo.	

Copia i dati della tabella da un account a un altro

Operazione	Description	Competenze richieste
Ottieni credenziali temporanee per i ruoli IAM.	Ottieni credenziali temporanee per il ruolo IAM creato nell'account di origine. Ottieni credenziali temporanee per il ruolo IAM creato nell'account di destinazione. Un modo per ottenere le credenziali temporanee per il ruolo IAM consiste nell'utilizzare AWS STS dalla CLI di AWS. `aws sts assume-role --role-arn arn:aws:i`	Sviluppatore di app, tecnico addetto alla migrazione

Operazione	Description	Competenze richieste
	`am::<account-id>:r ole/<role-name> -- role-session-name <session-name> -- profile <profile-name>` Utilizza il profilo AWS appropriato (corrispondente all'account di origine o di destinazione). Per ulteriori informazioni sui diversi modi per ottenere credenziali temporanee, consulta quanto segue: • Riferimento all'API AWS Security Token Service • Ottenere le credenziali del ruolo IAM per l'accesso alla CLI	

Operazione	Description	Competenze richieste
Inizializza i client DynamoDB per l'accesso a DynamoDB di origine e destinazione.	Inizializza i client DynamoDB, forniti dall'SDK AWS, per le tabelle DynamoDB di origine e di destinazione. • Per il client DynamoDB di origine, utilizza le credenziali temporanee recuperate dall'account di origine. • Per il client DynamoDB di destinazione, utilizza le credenziali temporanee recuperate dall'account di destinazione. Per ulteriori informazioni su come effettuare richieste utilizzando credenziali temporanee IAM, consulta la documentazione AWS.	Sviluppatore di app

Operazione	Description	Competenze richieste
Elimina e ricrea la tabella di destinazione.	Elimina e ricrea la tabella DynamoDB di destinazione (insieme agli indici) nell'acco unt di destinazione, utilizzando il client DynamoDB dell'acco unt di destinazione. L'eliminazione di tutti i record da una tabella DynamoDB è un'operazione costosa perché richiede il provisioning. WCUs L'eliminazione e la ricreazio ne della tabella evitano questi costi aggiuntivi. È possibile aggiungere indici a una tabella dopo averla creata, ma ciò richiede 2-5 minuti in più. La creazione di indici durante la creazione della tabella, passando la raccolta indexes alla chiamata, è più efficiente. `createTab le`	Sviluppatore di app

Operazione	Description	Competenze richieste
Eseguite la copia della tabella.	Ripetere i passaggi seguenti fino alla copia di tutti i dati: • Esegui una scansione della tabella nell'account di origine, utilizzando il client DynamoDB di origine. Ogni scansione DynamoDB recupera solo 1 MB di dati dalla tabella, quindi è necessario ripetere questa operazione fino alla lettura di tutti gli elementi o record. • Per ogni set di elementi scansionati, scrivi gli elementi nella tabella dell'account di destinazione, con il client DynamoDB di destinazione, utilizzando la `BatchWriteItem` chiamata nell'SDK AWS per DynamoDB. Ciò riduce il numero di `PutItem` richieste effettuate a DynamoDB. • `BatchWriteItem` ha un limite di 25 scritture o inserimenti o fino a 16 MB. È necessario aggiungere la logica per accumulare gli elementi scansionati contando fino a 25 prima di chiamare. `BatchWriteItem BatchWrit`	Sviluppatore di app

Operazione	Description	Competenze richieste
	`eItem` restituisce un elenco di elementi che non è stato possibile copiare correttamente. Utilizzando questo elenco, aggiungi la logica di ripetizione per eseguire un'altra `BatchWriteItem` chiamata solo con gli elementi che non hanno avuto successo. Per ulteriori informazioni, consulta l'implementazione di riferimento in C# (per eliminare, creare e popolare tabelle) nella sezione Allegati. È inoltre allegato un file JSON (JavaScript Object Notation) di configurazione della tabella di esempio.	

Risorse correlate

- Documentazione di Amazon DynamoDB
- Creazione di un utente IAM nel tuo account AWS
- AWS SDKs
- Utilizzo di credenziali temporanee con risorse AWS

Informazioni aggiuntive

Questo modello è stato implementato utilizzando C# per copiare una tabella DynamoDB con 200.000 elementi (dimensione media degli elementi di 5 KB e dimensione della tabella di 250 MB). La tabella

DynamoDB di destinazione è stata configurata con una capacità assegnata di 4000 e 4000. RCUs WCUs

L'operazione completa di copia della tabella (dall'account di origine all'account di destinazione), inclusa l'eliminazione e la ricreazione della tabella, ha richiesto 5 minuti. Capacità totale delle unità consumate: 30.000 RCUs e circa 400.000. WCUs

Per ulteriori informazioni sulle modalità di capacità di DynamoDB, consulta la modalità di capacità di lettura/scrittura nella documentazione AWS.

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Crea report dettagliati su costi e utilizzo per Amazon RDS e Amazon Aurora

Lakshmanan Lakshmanan e Sudarshan Narasimhan, Amazon Web Services

Riepilogo

Questo modello mostra come tenere traccia dei costi di utilizzo per i cluster Amazon Relational Database Service (Amazon RDS) o Amazon Aurora configurando tag di allocazione dei costi definiti dall'utente. Puoi utilizzare questi tag per creare report dettagliati su costi e utilizzo in AWS Cost Explorer per cluster su più dimensioni. Ad esempio, puoi tenere traccia dei costi di utilizzo a livello di team, progetto o centro di costo e quindi analizzare i dati in Amazon Athena.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Una o più istanze Amazon RDS o Amazon Aurora

Limitazioni

Per le restrizioni relative ai tag, consulta la AWS Billing User Guide.

Architecture

Stack tecnologico Target

- Amazon RDS o Amazon Aurora
- AWSReport di costi e utilizzo
- AWS Cost Explorer
- Amazon Athena

Flusso di lavoro e architettura

Il flusso di lavoro di etichettatura e analisi consiste nei seguenti passaggi:

1. Un ingegnere dei dati, un amministratore di database o un amministratore AWS crea tag di allocazione dei costi definiti dall'utente per i cluster Amazon RDS o Aurora.

2. Un amministratore AWS attiva i tag.
3. I tag inviano i metadati ad AWS Cost Explorer.
4. Un ingegnere dei dati, un amministratore di database o un amministratore AWS crea un rapporto mensile sull'allocazione dei costi.
5. Un ingegnere dei dati, un amministratore di database o un amministratore AWS analizza il report mensile di allocazione dei costi utilizzando Amazon Athena.

Il diagramma seguente mostra come applicare i tag per tenere traccia dei costi di utilizzo per le istanze Amazon RDS o Aurora.

Il seguente diagramma di architettura mostra come il report di allocazione dei costi è integrato con Amazon Athena per l'analisi.

Il report mensile di allocazione dei costi viene archiviato in un bucket Amazon S3 specificato dall'utente. Quando configuri Athena con il CloudFormation modello AWS, come descritto nella sezione Epics, il modello fornisce diverse risorse aggiuntive, tra cui un crawler AWS Glue, un database AWS Glue, un evento Amazon Simple Notification System (Amazon SNS), funzioni AWS Lambda e ruoli AWS Identity and Access Management (IAM) per le funzioni Lambda. Man mano che nuovi file di dati sui costi arrivano nel bucket S3, vengono utilizzate notifiche di eventi per inoltrare questi file a una funzione Lambda per l'elaborazione. La funzione Lambda avvia un crawler job di AWS Glue per creare o aggiornare la tabella nel catalogo dati di AWS Glue. Questa tabella viene quindi utilizzata per interrogare i dati in Athena.

Tools (Strumenti)

- Amazon Athena è un servizio di query interattivo che semplifica l'analisi dei dati in Amazon S3 utilizzando SQL standard.
- Amazon Aurora è un motore di database relazionale completamente gestito creato per il cloud e compatibile con MySQL e PostgreSQL.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS.

- AWS CloudFormation è un servizio Infrastructure as Code (IaC) che consente di modellare, fornire e gestire facilmente risorse AWS e di terze parti.
- AWS Cost Explorer ti aiuta a visualizzare e analizzare i costi e l'utilizzo di AWS.

Epiche

Crea e attiva tag per il tuo cluster Amazon RDS o Aurora

Operazione	Description	Competenze richieste
Crea tag di allocazione dei costi definiti dall'utente per il tuo cluster Amazon RDS o Aurora.	Per aggiungere tag a un cluster Amazon RDS o Aurora nuovo o esistente, segui le istruzioni in Aggiungere, elencare e rimuovere tag nella Guida per l'utente di Amazon Aurora. **Note** Per informazioni su come configurare un cluster Amazon Aurora, consulta le istruzioni per MySQL e PostgreSQL nella Guida per l'utente di Amazon Aurora.	Amministratore AWS, ingegnere dei dati, DBA
Attiva i tag di allocazione dei costi definiti dall'utente.	Segui le istruzioni in Attivazione dei tag di allocazione dei costi definiti dall'utente nella AWS Billing User Guide.	Amministratore AWS

Crea report su costi e utilizzo

Operazione	Description	Competenze richieste
Crea e configura report sui costi e sull'utilizzo per i tuoi cluster.	1. Accedi alla Console di gestione AWS e apri la console di fatturazione AWS. 2. Nel riquadro di navigazione a sinistra, scegli Report su costi e utilizzo. 3. Selezionare Create report (Crea report). 4. Fornisci un nome per il rapporto, mantieni le impostazioni predefinite per le altre opzioni, quindi scegli Avanti. 5. Scegli Configura e fornisci i dettagli di un bucket S3 esistente. Puoi anche scegliere di creare un nuovo bucket S3 da questa schermata. Scegli Next (Successivo). 6. Verifica la politica predefini ta che verrà applicata al tuo bucket, seleziona la casella di controllo di conferma, quindi scegli Salva. 7. Per il prefisso del percorso del report, specifica il prefisso che desideri aggiungere al nome del report.	Proprietario dell'app, amministratore AWS, DBA, General AWS, ingegnere dei dati

Operazione	Description	Competenze richieste
	8. Per la granularità temporale , scegli su base oraria, giornaliera o mensile, a seconda della frequenza con cui desideri che i dati vengano raccolti per il rapporto. 9. Per il controllo delle versioni del rapporto, scegli se desideri che le nuove versioni del rapporto vengano create separatam ente o sovrascrivi il rapporto esistente con ogni versione. 10. Per Abilita l'integrazione dei dati dei report per, scegli Amazon Athena. Verifica che il tipo di compressione sia impostato su Parquet. 11. Scegli Next (Successivo). 12. Rivedi le impostazioni del rapporto, quindi scegli Revisione e completa. I dati saranno disponibili in 24 ore.	

Analizza i dati dei report su costi e utilizzo

Operazione	Description	Competenze richieste
Analizza i dati del rapporto sui costi e sull'utilizzo.	1. Configura e usa Athena per analizzare i dati del report.	Proprietario dell'app, amministratore AWS, DBA,

Operazione	Description	Competenze richieste
	Per istruzioni, consulta la sezione Query Cost and Usage Reports utilizzando Amazon Athena nella AWS Cost and Usage Reports User Guide. Ti consigliamo di utilizzare il CloudFormation modello AWS fornito da Athena. 2. Esegui le interrogazioni Athena. Ad esempio, è possibile utilizzare la seguente query SQL per verificare lo stato dell'aggiornamento dei dati. `select status from cost_and_usage_data_status` Per ulteriori informazioni, consulta Esecuzione di query Amazon Athena nella AWS Cost and Usage Reports User Guide. **Note** Quando esegui la query SQL, assicurati che il database corretto sia seleziona	General AWS, ingegnere dei dati

Operazione	Description	Competenze richieste
	to dall'elenco a discesa.	

Risorse correlate

Riferimenti

- Configurazione di Athena utilizzando CloudFormation modelli AWS (consigliata)
- Configurazione manuale di Athena
- Esecuzione di query Amazon Athena
- Caricamento dei dati dei report su altre risorse

Tutorial e video

- Analizza i report su costi e utilizzo con Amazon Athena (video) YouTube

Distribuisci istanze di cluster di failover di SQL Server su Amazon e EC2 Amazon FSx utilizzando Terraform

Mark Hudson e Matt Burgess, Amazon Web Services

Riepilogo

Questo modello utilizza Terraform per distribuire istanze di cluster di failover di SQL Server (FCIs) su nodi Windows Server Failover Cluster (WSFC) su Amazon Elastic Compute Cloud (Amazon). EC2 Inoltre, il modello utilizza lo storage FSx condiviso di Amazon per dati e file di registro.

Quando si esegue la migrazione dei database SQL Server AWS, la prima scelta è Amazon RDS for SQL Server. Tuttavia, a volte Amazon RDS for SQL Server non è adatto e SQL Server deve essere distribuito su EC2 Amazon in un'architettura ad alta disponibilità. In questa soluzione, SQL Server viene installato su FCIs nodi WSFC.

Il modulo Terraform incluso in questo modello fornisce fino a due istanze di Amazon EC2 SQL Server. Un file system Amazon FSx for Windows File Server funge da testimone del quorum e archivia dati e file di registro condivisi. Indipendentemente dal numero di istanze configurate, i nodi di istanze di SQL Server creeranno e si uniranno sempre a un cluster FCI per garantire la parità ambientale. (In genere, un'istanza è configurata per lo sviluppo e due istanze per gli ambienti di produzione). Per le configurazioni che utilizzano due nodi per l'elevata disponibilità, viene fornito un Network Load Balancer interno. Il Network Load Balancer utilizza una sonda di salute configurata sul cluster FCI per identificare quale nodo è il principale.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Amazon Virtual Private Cloud (Amazon VPC) con due sottoreti in zone di disponibilità separate.
- Set di opzioni DHCP Amazon VPC. Configura il nome di dominio in modo che si risolva nel nome di dominio Active Directory e i server dei nomi di dominio e NetBIOS in modo che puntino ai controller di dominio Active Directory. Per ulteriori informazioni, consulta Configurazione VPC in Informazioni aggiuntive.
- AWS Directory Service for Microsoft Active Directory (AWS Managed Microsoft AD).
- Amazon Machine Image (AMI) personalizzata. Per maggiori dettagli, consulta Configurazione AMI in Informazioni aggiuntive.

- Bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) contenente l'immagine ISO di SQL Server. Questo prerequisito è richiesto solo se si utilizza EC2 Image Builder con il file `component.yaml` fornito per creare l'AMI personalizzata.
- AWS Key Management Service (AWS KMS) chiave di crittografia.
- Per impostazione predefinita, SQL Server viene installato utilizzando un codice prodotto della Developer Edition. I sistemi di produzione dovrebbero utilizzare un codice prodotto valido passato al modulo dalla variabile pertinente.

Limitazioni

- Questa soluzione richiede AWS Managed Microsoft AD. Tuttavia, se preferisci, puoi invece utilizzare un'implementazione di Active Directory autogestita. A tale scopo, modifica il modulo Amazon FSx Terraform incluso per rimuovere l'`active_directory_id`attributo. Quindi, aggiungi i quattro attributi necessari per l'autogestione di Active Directory, come mostrato nella documentazione di Terraform.
- SQL Server è configurato per utilizzare l'autenticazione in modalità mista. Se preferisci, puoi utilizzare l'autenticazione solo per Windows. A tale scopo, nello script di dati utente fornito, rimuovete i `/SAPWD` parametri `/SECURITYMODE` and forniti al `setup.exe` comando. È possibile rimuovere il `sql_accounts.tf` file e modificare il `instances.tf` file per rimuovere la `sql_sa_password` voce.
- Quando si elimina un cluster distribuito, è necessario rimuovere i corrispondenti oggetti del computer virtuale e i singoli oggetti del computer in Active Directory. Per rimuovere gli oggetti, utilizzare gli strumenti di amministrazione di Active Directory.
- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità regionale, vedi AWS Servizi per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Versioni del prodotto

Questa soluzione è stata testata con le seguenti versioni:

- Windows Server 2019
- SQL Server 2019
- Terraform v0.13.0

Architecture

Stack tecnologico di origine

- SQL Server

Stack tecnologico di destinazione

- SQL Server FCI su nodi WSFC con Amazon EC2
- File server Amazon FSx per Windows
- Bucket Amazon S3
- Gestione dei segreti AWS
- AWS Managed Microsoft AD
- AWS KMS
- AWS Identity and Access Management (IAM)

Architettura Target

Il diagramma seguente mostra l'architettura di questa soluzione.

Il diagramma mostra:

- Un ruolo IAM che fornisce alle EC2 istanze l'accesso a AWS KMS Secrets Manager
- Due nodi SQL Server distribuiti su EC2 istanze Amazon in sottoreti private su due zone di disponibilità
- Un Network Load Balancer per facilitare le connessioni all'istanza attiva di SQL Server (non distribuito durante la configurazione di un cluster a nodo singolo)
- File system Amazon FSx per Windows File Server distribuito in entrambe le sottoreti private per lo storage condiviso dai nodi di SQL Server
- Secrets Manager per l'archiviazione delle credenziali e della configurazione di Active Directory e SQL Server
- Bucket Amazon S3 per l'archiviazione dell'immagine di installazione di SQL Server
- AWS Managed Microsoft AD per l'autenticazione di Windows
- AWS KMS per creare la chiave di crittografia

Automazione e scalabilità

È possibile automatizzare l'implementazione dell'architettura di destinazione utilizzando i moduli Terraform presenti nel GitHub repository. È necessario modificare il `terraform.tfvars` file per includere valori variabili specifici del proprio ambiente. Il bucket Amazon S3, AWS Managed Microsoft AD i componenti, la chiave di AWS KMS crittografia e alcuni segreti sono prerequisiti per questa distribuzione e non sono inclusi nel codice Terraform.

Tools (Strumenti)

Servizi AWS

- AWS Directory Service for Microsoft Active Directoryconsente ai carichi di lavoro e alle risorse compatibili con le directory di AWS utilizzare Microsoft Active Directory in. Cloud AWS In questo modello, AWS Managed Microsoft AD viene utilizzato per l'autenticazione di Windows Server e SQL Server e per il DNS.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente. In questo modello, le istanze del cluster di failover di SQL Server vengono installate sulle istanze Amazon EC2 .
- EC2 Image Builder consente di automatizzare la creazione, la gestione e l'implementazione di immagini server personalizzate.
- Amazon FSx for Windows File Server fornisce storage condiviso completamente gestito su Windows Server. In questo modello, FSx per Windows File Server fornisce lo storage condiviso per i dati e i file di registro di SQL Server e il quorum witness.
- AWS Identity and Access Management (IAM) consente di gestire in modo sicuro l'accesso alle AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati. In questo modello, viene utilizzato per crittografare i segreti di Secrets Manager, lo storage SQL Server sui volumi Amazon Elastic Block Store (Amazon EBS) e il file system FSx per Windows File Server.
- Con Gestione dei segreti AWS puoi sostituire le credenziali nel codice,incluse le password, con una chiamata API a Secrets Manager in modo da recuperare il segreto a livello di codice. In questo modello, le credenziali di Active Directory per l'installazione e l'esecuzione di SQL Server, le credenziali `sa` utente e le informazioni di connessione al database vengono archiviate in Secrets Manager.

- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati. Questo modello utilizza un bucket Amazon S3 per archiviare l'immagine di installazione di SQL Server.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.

Altri strumenti

- Microsoft SQL Server FCIs vengono installati nei nodi del cluster di Windows Server. Inoltre, possono essere installati su più sottoreti. In questo modello, le istanze FCI di SQL Server vengono installate tra i nodi WSFC.
- Terraform è uno strumento Infrastructure as Code (IaC) che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud. In questo modello, Terraform viene utilizzato per creare le risorse e configurare le istanze FCI di SQL Server.
- Windows Server Failover Clustering offre funzionalità di infrastruttura che supportano l'elevata disponibilità di applicazioni server ospitate come SQL Server. In questo modello, i nodi FCI utilizzano la funzionalità WSFC per fornire un'elevata disponibilità locale attraverso la ridondanza a livello di istanza.

Archivio di codici

Il codice per questo pattern è disponibile nel repository GitHub cluster-amazon-elastic-compute-cloud-amazon-fsx-microsoft -sql-server. Nel repository sono disponibili le seguenti risorse:

- Un `README.md` file che fornisce una panoramica della soluzione e informazioni aggiuntive sull'installazione e l'utilizzo
- Un set base di file di configurazione Terraform e un modulo FSx specifico di Amazon per fornire i componenti per questo modello
- Uno script di configurazione dell'istanza che viene eseguito come script di dati EC2 utente Amazon
- Un file `component.yaml` che Image Builder può utilizzare per creare un'AMI personalizzata

Best practice

Sicurezza e applicazione di patch

- Le installazioni e le configurazioni dei prerequisiti AMI sono i requisiti minimi per distribuire i cluster FCI di SQL Server. Potrebbero essere necessari software e configurazioni aggiuntivi per conformarsi agli standard e ai requisiti di sicurezza dell'organizzazione.
- Dopo la distribuzione, applica patch a Windows su base continuativa. Puoi applicare direttamente le patch alle istanze in esecuzione o creare una nuova AMI con le ultime patch di Windows e sostituire le istanze (una alla volta) utilizzando la nuova AMI. AWS rilascia AMIs ogni mese nuovi Windows contenenti le patch, i driver e gli agenti di lancio più recenti del sistema operativo. Ti consigliamo di verificare la disponibilità dell'AMI più recente quando avvii nuove istanze o quando crei immagini personalizzate.
- Le EC2 istanze Amazon sono configurate per consentire tutto il traffico in uscita. Se distribuito in un ambiente di produzione, è necessario stabilire regole in uscita nel gruppo di sicurezza per limitare questo traffico alle destinazioni richieste.
- Il file system FSx per Windows File Server può registrare automaticamente i registri di controllo per la condivisione di file e l'accesso a file e cartelle e inviarli alla destinazione desiderata, se questo è un requisito dell'ambiente.
- Ruota i segreti di Secrets Manager automaticamente a intervalli regolari. Per la coppia di chiavi di EC2 istanze Amazon, prendi in considerazione una soluzione di rotazione automatizzata come descritto in Come usare AWS Secrets Manager per archiviare e ruotare in modo sicuro le coppie di chiavi SSH. Per le credenziali di Active Directory e le `sa` credenziali segrete di SQL Server, configura la rotazione automatica in base alle tue politiche di gestione delle password.

Gestione di Active Directory

- Come parte del cluster FCI, Windows genera un Computer Name Object (CNO) in Active Directory. Il CNO risponde alle richieste DNS e inoltra il traffico al nodo SQL attivo. Non è consigliabile utilizzare questo DNS fornito da Active Directory. Il TTL è troppo alto per consentire un tempo di failover ragionevole e spesso impiega fino a 5 minuti per riflettere il nuovo indirizzo IP primario. Al contrario, per le installazioni ad alta disponibilità, il Network Load Balancer interno è configurato per il failover entro 30 secondi.
- Per creare il cluster è necessario un amministratore di dominio Active Directory. Questo requisito è dovuto alle autorizzazioni elevate richieste per creare gli oggetti del cluster e modificare le autorizzazioni in Active Directory. Tuttavia, i servizi SQL Server non devono essere eseguiti come amministratore di dominio. Pertanto, si consiglia di creare un secondo utente Active Directory per questo scopo. Tuttavia, è possibile eliminare questo utente se i servizi verranno eseguiti come

utente amministratore di dominio. In tal caso, l'utente amministratore di dominio deve essere aggiunto al gruppo di amministratori di Active Directory creato come parte di questo modello.

Epiche

Configura le credenziali del cluster

Operazione	Description	Competenze richieste
Creare gruppi Active Directory.	In AWS Managed Microsoft AD, crea i seguenti gruppi: • Gruppo di amministratori del cluster: questo gruppo verrà aggiunto al gruppo Administrators locale su ogni nodo del cluster. • Gruppo di desktop remoto del cluster: questo gruppo verrà aggiunto al gruppo locale Remote Desktop Users su ogni nodo del cluster. Per maggiori dettagli, consulta Creazione di un AWS Managed Microsoft AD gruppo nella AWS documentazione.	Amministratore AD
Crea utenti Active Directory.	In AWS Managed Microsoft AD, crea i seguenti utenti • Utente amministratore di dominio: utilizza questo account per creare il cluster. • Utente di dominio: i servizi di SQL Server utilizzeranno	Amministratore AD

Operazione	Description	Competenze richieste
	questo account per l'esecuzione. Aggiungi questo utente al gruppo di amministratori del cluster creato nell'attività precedente. Per maggiori dettagli, consulta Creazione di un AWS Managed Microsoft AD utente nella AWS documentazione.	
Aggiungi le credenziali di Active Directory ai segreti.	Usa Secrets Manager per creare quattro segreti per archiviare le seguenti informazioni: • Nome utente dell'amministratore di dominio • Password dell'utente dell'amministratore di dominio • Nome utente dell'utente del dominio • Password dell'utente del dominio Per maggiori dettagli, consulta Creare un Gestione dei segreti AWS segreto nella AWS documentazione.	Amministratore AWS

Prepara l'ambiente

Operazione	Description	Competenze richieste
Crea l'AMI di Windows.	Crea un'AMI Windows personalizzata che includa il software e le configura zioni prerequisiti. Per ulteriori dettagli, vedere Informazioni aggiuntive.	Amministratore AWS, AWS DevOps
Installa Terraform.	Per installare Terraform, segui le istruzioni sul sito Web Terraform.	AWS DevOps
Clonare il repository.	Clona il repository di questo pattern. Per maggiori dettagli, consulta Clonazione di un repository sul sito web. GitHub	AWS DevOps

Installa il cluster

Operazione	Description	Competenze richieste
Modifica le variabili Terraform.	Aggiorna il `terraform .tfvars` file fornito per impostare tutte le variabili sui valori appropriati per il tuo ambiente. Ad esempio, aggiorna le `domain_group_rdp_u sers` variabili `domain_gr oup_administrators` and per utilizzare il nome di dominio Active Directory	AWS DevOps

Operazione	Description	Competenze richieste
	e il nome dei gruppi Active Directory creati in precedenza.	
Inizializza Terraform.	Per vedere la distribuzione proposta, vai alla radice del repository. Usa l'interfaccia a riga di comando (CLI) Terraform per `terraform init` eseguire e poi eseguire. `terraform plan`	AWS DevOps
Distribuisci risorse.	Per distribuire il cluster SQL e le risorse associate, utilizza la CLI Terraform per l'esecuzione. `terraform apply`	AWS DevOps, amministratore AWS

Operazione	Description	Competenze richieste
Convalida la distribuzione.	Per convalidare la distribuzione, attenersi alla seguente procedura: 1. Connect a una delle EC2 istanze Windows Amazon distribuite utilizzando Remote Desktop. 2. Per ulteriori dettagli, vedi Connect alla tua istanza di Windows usando RDP nella AWS documentazione. 3. Apri Windows Failover Cluster Manager. Verifica che il cluster sia stato creato e che il ruolo di SQL Server sia stato creato e sia in esecuzione. 4. Per testare la connettività e l'autenticazione con SQL Server, utilizza uno strumento di database come SQL Server Management Studio per connetterti all'endpoint SQL Server. (Il valore dell'endpoint è memorizzato in Secrets Manager.) Per maggiori dettagli, consulta Connect to Microsoft SQL Server on Amazon EC2 nella AWS documentazione.	Amministratore di sistema DBA, AWS

risoluzione dei problemi

Problema	Soluzione
Il provisioning di Terraform è stato completato ma Windows Failover Cluster Manager non mostra che è stato creato un cluster o che il cluster è in uno stato non utilizzabile.	L'intera installazione delle risorse e la configurazione dei cluster possono richiedere 45-60 minuti. Una volta completato Terraform, lo script dei dati utente deve essere eseguito fino al completamento, il che richiede più riavvii. Per monitorare l'avanzamento, è possibile utilizzare la `Checkpoints` directory nell'`C:\`unità e accedere all'installazione di SQL Server. `C:\Program Data\Microsoft SQL Server\150\Log` Al termine, il messaggio Installazione completata è disponibile nel `C:\ProgramData\Amazon\EC2-Windows\Launch\Log\UserdataExecution.log` file.
Dopo aver effettuato il provisioning di un cluster funzionante, l'utilizzo di Terraform per eliminare e ricreare il cluster non ha esito positivo. Terraform viene completato, ma il cluster non è configurato correttamente.	Parte del processo di provisioning prevede la registrazione di macchine e oggetti virtuali su Active Directory e Active Directory DNS. Se esistono nomi di computer per i nodi del EC2 cluster Amazon e il nodo del cluster, FCI non può inizializzarsi correttamente e il provisioning fallirà. Per risolvere questo problema, procedi nel seguente modo: 1. Elimina i EC2 nodi Amazon, il computer virtuale del cluster e l'ID del cluster negli utenti e nei computer di Active Directory. 2. Elimina la voce DNS per il computer virtuale del cluster in Active Directory DNS. 3. Esegui il seguente comando per eliminare la risorsa Terraform, stringa casuale, id

Problema	Soluzione
	del cluster. `terragrunt destroy -target=random_string.cluster_id` Questa azione eliminerà le EC2 istanze Amazon esistenti. 4. Esegui `terraform apply` e aspettati le seguenti tre nuove risorse: due EC2 istanze Amazon FCI e 1 ID cluster di stringhe casuali.

Risorse correlate

AWS documentazione

- Crea immagini personalizzate con Image Builder
- Crea una chiave KMS
- Creazione di un bucket generico
- Creazione di una politica chiave
- Creando il tuo AWS Managed Microsoft AD

Informazioni aggiuntive

Informazioni sul modulo Terraform

Questo modulo utilizza una combinazione di configurazione AMI e configurazione dei dati utente per ottenere un buon mix di tempi di provisioning e stabilità. Durante il provisioning, Windows richiede più riavvii e attese. È stato implementato un metodo di controllo per proteggere dai loop infiniti durante il riavvio dei dati utente persistenti. I dati utente sono configurati per essere persistenti. Pertanto, gli script di configurazione dei dati utente devono e devono continuare a essere sviluppati per essere idempotenti. Idempotency semplifica il processo di aggiornamento, consentendo la sostituzione delle istanze durante un ciclo di aggiornamento senza la configurazione manuale per ricongiungersi o ricreare i cluster FCI.

Stringhe di connessione e clustering di failover di SQL Server

Il modulo pubblicherà un segreto contenente l'indirizzo dell'endpoint da utilizzare nelle stringhe di connessione per questo database. Il nome segreto segue il formato. `{environment_name}/sqlserver/{cluster_name}/endpoint` Per le installazioni in cui viene utilizzato un solo nodo, puoi aspettarti che questo sia l'indirizzo IP dell'interfaccia SQL Server dell' EC2 istanza Amazon. Per le installazioni ad alta disponibilità (due istanze), puoi aspettarti che questo sia il nome DNS del Network Load Balancer interno.

Il clustering virtuale di failover IPs non è supportato in questo modulo. Un IP virtuale deve rimanere nella stessa sottorete per funzionare. Nel AWS, una singola sottorete non può estendersi su più zone di disponibilità. Pertanto, l'uso del virtuale IPs eliminerebbe la possibilità per questo modulo di essere considerato altamente disponibile.

A ogni EC2 istanza Amazon vengono assegnati tre indirizzi IP privati. Il loro utilizzo è il seguente:

- IP primario per il traffico di rete: l'IP di origine per il traffico in uscita.
- Comunicazioni FCI: utilizzate per mantenere lo stato e la sincronizzazione del cluster di failover.
- SQL Server (porta TCP 1433): ascolta e ascolta anche il traffico heartbeat per determinare quale istanza è principale.

Configurazione VPC

I prerequisiti elencano un set di opzioni DHCP configurato per utilizzare Active Directory per la risoluzione DNS. Tuttavia, questo prerequisito non è un requisito rigido. Il requisito fondamentale è che le EC2 istanze devono essere in grado di risolvere il nome di dominio Active Directory. È possibile soddisfare questo requisito in altri modi, ad esempio utilizzando Amazon Route 53 Resolver gli endpoint. Per ulteriori informazioni, consulta Integrazione della risoluzione DNS del servizio di directory con Amazon Route 53 Resolvers (post di blog).AWS

Configurazione AMI

L'AMI utilizzata in questo modello deve contenere il software e le configurazioni prerequisiti seguenti:

1. Scarica ed espandi i file di installazione di SQL Server 2019 in. `C:\SQL_Install_media`
2. Installa le seguenti funzionalità di Windows:
   - `Install-WindowsFeature Failover-Clustering`
   - `Install-WindowsFeature RSAT-AD-PowerShell`
   - `Install-WindowsFeature RSAT-AD-Tools`

- `Install-WindowsFeature RSAT-Clustering-Mgmt`
- `Install-WindowsFeature RSAT-Clustering-PowerShell`
- `Install-WindowsFeature RSAT-Clustering-CmdInterface`

3. Disattiva il firewall di Windows come segue:
   - `Get-NetFirewallProfile | Set-NetFirewallProfile -Enabled False`
4. Abilita il metodo di autenticazione CredSSP (`<domain>`sostituiscilo con il nome di dominio Windows dell'organizzazione) come segue:
   - `Enable-WSManCredSSP -Role "Server" -Force`
   - `Enable-WSManCredSSP -Role "Client" -DelegateComputer *.<domain>.com -Force`
5. Imposta le seguenti chiavi di registro:
   - Consenti le credenziali di autenticazione NTLM:
     - `HKLM:\Software\Policies\Microsoft\Windows\CredentialsDelegation`
       - Valore: `AllowFreshCredentialsWhenNTLMOnly`
       - Valore: `1`
       - Tipo: `REG_DWORD`
   - Consenti ai computer del dominio locale di utilizzare NTLM da PowerShell:
     - Percorso: `HKLM:\Software\Policies\Microsoft\Windows\CredentialsDelegation\AllowFreshCredentialsWhenNTLMOnly`
       - Valore: `1`
       - Valore: `wsman/*.<domain>.com`
       - Tipo: `REG_SZ`
6. Configura la PowerShell Galleria come segue:
   - `[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12`
   - `Install-PackageProvider -Name NuGet -MinimumVersion 2.8.5.201 -Force`
   - `Set-PSRepository -Name PSGallery -InstallationPolicy Trusted`
7. Installa i seguenti PowerShell moduli Windows*:
   - `Install-Module -Name ComputerManagementDsc`
   - `Install-Module -Name FailOverClusterDsc`
   - `Install-Module -Name PSDscResources`

- `Install-Module -Name xSmbShare`
- `Install-Module -Name xActiveDirectory`
- `Install-Module -Name SqlServer`

Per utilizzare Image Builder per creare l'AMI, segui le istruzioni in Creare una pipeline di immagini utilizzando la procedura guidata della console di Image Builder nella documentazione di EC2 Image Builder. Per creare il componente della ricetta con i prerequisiti precedenti, procedi nel seguente modo:

1. Scarica il file component.yaml dalla cartella del `ami` repository. GitHub
2. Copiate il contenuto in un nuovo componente Image Builder.
3. Aggiorna i seguenti segnaposto con le tue informazioni:
   - `<domain>`— Il tuo nome di dominio Active Directory
   - `<bucket_name>`— Nome del bucket Amazon S3 che contiene l'immagine SQL Server

Emula gli array PL/SQL associativi Oracle in Amazon Aurora PostgreSQL e Amazon RDS per PostgreSQL

Rajkumar Raghuwanshi, Bhanu Ganesh Gudivada e Sachin Khanna, Amazon Web Services

Riepilogo

Questo modello descrive come emulare gli array PL/SQL associativi Oracle con posizioni di indice vuote in ambienti Amazon Aurora PostgreSQL e Amazon RDS for PostgreSQL. Descrive inoltre alcune delle differenze tra gli array PL/SQL associativi Oracle e gli array PostgreSQL per quanto riguarda il modo in cui ciascuno gestisce le posizioni degli indici vuoti durante le migrazioni.

Forniamo un'alternativa PostgreSQL all'`aws_oracle_ext`utilizzo di funzioni per la gestione delle posizioni di indice vuote durante la migrazione di un database Oracle. Questo modello utilizza una colonna aggiuntiva per memorizzare le posizioni degli indici e mantiene la gestione degli array sparsi da parte di Oracle incorporando funzionalità PostgreSQL native.

Oracle

In Oracle, le raccolte possono essere inizializzate come vuote e popolate utilizzando il metodo `EXTEND` collection, che aggiunge elementi all'array. `NULL` Quando si lavora con array PL/SQL associativi indicizzati da`PLS_INTEGER`, il `EXTEND` metodo aggiunge `NULL` elementi in sequenza, ma gli elementi possono anche essere inizializzati in posizioni di indice non sequenziali. Qualsiasi posizione dell'indice che non sia inizializzata in modo esplicito rimane vuota.

Questa flessibilità consente strutture di array sparse in cui gli elementi possono essere popolati in posizioni arbitrarie. Quando si esegue un'iterazione tra le raccolte utilizzando un `FOR LOOP` with `FIRST` e un `LAST` bounds, vengono elaborati solo gli elementi inizializzati (con `NULL` o con un valore definito), mentre le posizioni vuote vengono saltate.

PostgreSQL (Amazon Aurora e Amazon RDS)

PostgreSQL gestisce i valori vuoti in modo diverso dai valori. `NULL` Memorizza i valori vuoti come entità distinte che utilizzano un byte di archiviazione. Quando un array ha valori vuoti, PostgreSQL assegna posizioni di indice sequenziali proprio come valori non vuoti. Ma l'indicizzazione sequenziale richiede un'elaborazione aggiuntiva perché il sistema deve scorrere attraverso tutte le posizioni indicizzate, comprese quelle vuote. Ciò rende la creazione di array tradizionali inefficiente per set di dati sparsi.

AWS Schema Conversion Tool

Il AWS Schema Conversion Tool (AWS SCT) in genere gestisce le Oracle-to-PostgreSQL migrazioni utilizzando le funzioni. `aws_oracle_ext` In questo modello, proponiamo un approccio alternativo che utilizza le funzionalità native di PostgreSQL, che combina i tipi di array PostgreSQL con una colonna aggiuntiva per l'archiviazione delle posizioni degli indici. Il sistema può quindi eseguire iterazioni tra gli array utilizzando solo la colonna dell'indice.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Un utente AWS Identity and Access Management (IAM) con autorizzazioni di amministratore.
- Un'istanza compatibile con Amazon RDS o Aurora PostgreSQL.
- Una conoscenza di base dei database relazionali.

Limitazioni

- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Versioni del prodotto

Questo modello è stato testato con le seguenti versioni:

- Amazon Aurora PostgreSQL 13.3
- Amazon RDS per PostgreSQL 13.3
- AWS SCT 1.0.674
- Oracle12c EE 12.2

Architecture

Stack tecnologico di origine

- Database Oracle locale

Stack tecnologico Target

- Amazon Aurora PostgreSQL
- Amazon RDS per PostgreSQL

Architettura di destinazione

Il diagramma mostra:

- Un'istanza di database Amazon RDS for Oracle di origine
- Un' EC2 istanza Amazon AWS SCT per convertire le funzioni Oracle nell'equivalente di PostgreSQL
- Un database di destinazione compatibile con Amazon Aurora PostgreSQL

Tools (Strumenti)

Servizi AWS

- Amazon Aurora è un motore di database relazionale completamente gestito creato per il cloud e compatibile con MySQL e PostgreSQL.
- Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale in. Cloud AWS
- Amazon Relational Database Service (Amazon RDS) per Oracle ti aiuta a configurare, gestire e scalare un database relazionale Oracle in. Cloud AWS
- Amazon Relational Database Service (Amazon RDS) per PostgreSQL ti aiuta a configurare, gestire e scalare un database relazionale PostgreSQL in. Cloud AWS
- AWS Schema Conversion Tool (AWS SCT) supporta migrazioni di database eterogenee convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione.

Altri strumenti

- Oracle SQL Developer è un ambiente di sviluppo integrato che semplifica lo sviluppo e la gestione dei database Oracle nelle implementazioni tradizionali e basate sul cloud.
- pgAdmin è uno strumento di gestione open source per PostgreSQL. Fornisce un'interfaccia grafica che consente di creare, gestire e utilizzare oggetti di database. In questo modello, pgAdmin si connette all'istanza del database RDS per PostgreSQL e interroga i dati. In alternativa, puoi usare il client da riga di comando psql.

Best practice

- Verifica i limiti dei set di dati e gli scenari periferici.
- Prendi in considerazione l'implementazione della gestione degli errori per le condizioni out-of-bounds dell'indice.
- Ottimizza le query per evitare la scansione di set di dati sparsi.

Epiche

Comportamento degli array associativi Oracle (fonte)

Operazione	Description	Competenze richieste
Crea un PL/SQL blocco sorgente in Oracle.	Crea un PL/SQL blocco sorgente in Oracle che utilizza il seguente array associativo: `DECLARE TYPE country_codes IS TABLE OF VARCHAR2(100) INDEX BY pls_integer; cc country_codes; cc_idx NUMBER := NULL; BEGIN cc(7) := 'India'; cc(3) := 'UK'; cc(5) := 'USA';`	DBA

Operazione	Description	Competenze richieste

	`cc(0) := 'China';`
	`cc(-2) := 'Invalid'`
	`;`
	`dbms_output.put_li`
	`ne('cc_length:' ||`
	` cc.COUNT);`
	`IF (cc.COUNT > 0)`
	` THEN`
	`cc_idx :=`
	` cc.FIRST;`
	`FOR i IN`
	` 1..cc.COUNT LOOP`
	`dbms_outp`
	`ut.put_line('cc_id`
	`x:' || cc_idx || '`
	` country:' || cc(cc_idx`
	`));`
	`cc_idx :=`
	` cc.next(cc_idx);`
	`END LOOP;`
	`END IF;`
	`END;`	

Esegui il PL/SQL blocco.	Esegui il PL/SQL blocco sorgente in Oracle. Se ci sono degli spazi tra i valori di indice di un array associativo, in tali spazi non viene memorizza to alcun dato. Ciò consente al ciclo Oracle di scorrere solo attraverso le posizioni dell'indi ce.	DBA

Operazione	Description	Competenze richieste

Esamina l'output.	Cinque elementi sono stati inseriti nell'array (`cc`) a intervalli non consecutivi. Il conteggio degli array è mostrato nel seguente output:
	`cc_length:5`
	`cc_idx:-2 country:I nvalid`
	`cc_idx:0 country:China`
	`cc_idx:3 country:UK`
	`cc_idx:5 country:USA`
	`cc_idx:7 country:India`
		DBA

Comportamento dell'array associativo PostgreSQL (target)

Operazione	Description	Competenze richieste

Crea un PL/pgSQL blocco di destinazione in PostgreSQL.	Crea un PL/pgSQL blocco di destinazione in PostgreSQL che utilizza il seguente array associativo:
	`DO $$`
	`DECLARE`
	`cc character varying(100)[];`
	`cc_idx integer := NULL;`
	`BEGIN`
	`cc[7] := 'India';`
	`cc[3] := 'UK';`
	`cc[5] := 'USA';`
	`cc[0] := 'China';`
	`cc[-2] := 'Invalid' ;`
		DBA

Operazione	Description	Competenze richieste

	<pre> RAISE NOTICE
	 'cc_length: %',
	 ARRAY_LENGTH(cc, 1);
	    IF (ARRAY_LE
	NGTH(cc, 1) > 0) THEN
	        FOR i IN
	 ARRAY_LOWER(cc,
	 1)..ARRAY_UPPER(cc,
	 1)
	        LOOP
	            RAISE NOTICE
	 'cc_idx:% country:%',
	 i, cc[i];
	        END LOOP;
	    END IF;
	END;
	$$;</pre>	

Esegui il PL/pgSQL blocco.	Esegui il PL/pgSQL blocco di destinazione in PostgreSQL. Se ci sono degli spazi tra i valori dell'indice di un array associativo, nessun dato viene memorizzato in tali spazi. Ciò consente al ciclo Oracle di scorrere solo attraverso le posizioni dell'indice.	DBA

Operazione	Description	Competenze richieste

Esamina l'output.	La lunghezza dell'array è maggiore di 5 perché `NULL` è memorizzata negli spazi tra le posizioni dell'indice. Come mostrato nell'output seguente, il ciclo completa 10 iterazioni per recuperare 5 valori nell'array.
	`cc_length:10`
	`cc_idx:-2 country:I`
	`nvalid`
	`cc_idx:-1 country:<`
	`NULL>`
	`cc_idx:0 country:China`
	`cc_idx:1 country:<`
	`NULL>`
	`cc_idx:2 country:<`
	`NULL>`
	`cc_idx:3 country:UK`
	`cc_idx:4 country:<`
	`NULL>`
	`cc_idx:5 country:USA`
	`cc_idx:6 country:<`
	`NULL>`
	`cc_idx:7 country:India`
		DBA

Emula il comportamento degli array associativi Oracle

Operazione	Description	Competenze richieste
Crea un PL/pgSQL blocco di destinazione con un array e un tipo definito dall'utente.	Per ottimizzare le prestazioni e soddisfare le funzionalità di Oracle, possiamo creare un tipo definito dall'utente che memorizza sia le posizioni	DBA

Operazione	Description	Competenze richieste
	dell'indice che i dati corrispon denti. Questo approccio riduce le iterazioni non necessari e mantenendo associazioni dirette tra indici e valori. `DO $$ DECLARE cc country_codes[]; cc_append country_codes := NULL; i record; BEGIN cc_append.idx = 7; cc_append.val = 'India'; cc := array_append(cc, cc_append); cc_append.idx = 3; cc_append.val = 'UK'; cc := array_append(cc, cc_append); cc_append.idx = 5; cc_append.val = 'USA'; cc := array_append(cc, cc_append); cc_append.idx = 0; cc_append.val = 'China'; cc := array_append(cc, cc_append); cc_append.idx = -2; cc_append.val = 'Invalid'; cc := array_append(cc, cc_append);`	

Operazione	Description	Competenze richieste

	<pre> RAISE NOTICE
	 'cc_length: %',
	 ARRAY_LENGTH(cc, 1);
	    IF (ARRAY_LE
	NGTH(cc, 1) > 0) THEN
	        FOR i IN (
	            SELECT
	                *
	            FROM
	
	 unnest(cc)
	            ORDER BY
	                idx)
	                LOOP
	
	 RAISE NOTICE 'cc_idx:
	% country:%', i.idx,
	 i.val;
	                END
	 LOOP;
	    END IF;
	END;
	$$;</pre>	

Esegui il PL/pgSQL blocco.	Esegui il PL/pgSQL blocco bersaglio. Se ci sono degli spazi tra i valori dell'indice di un array associativo, in tali spazi non viene memorizza to alcun dato. Ciò consente al ciclo Oracle di scorrere solo attraverso le posizioni dell'indi ce.	DBA

Operazione	Description	Competenze richieste

Esamina l'output.	Come illustrato nell'outp ut seguente, il tipo definito dall'utente memorizza solo gli elementi di dati popolati, il che significa che la lunghezza dell'array corrispon de al numero di valori. Di conseguenza, L00P le iterazioni sono ottimizzate per elaborare solo i dati esistenti , eliminando la necessità di tenere traccia delle posizioni vuote.
	`cc_length:5`
	`cc_idx:-2 country:I nvalid`
	`cc_idx:0 country:China`
	`cc_idx:3 country:UK`
	`cc_idx:5 country:USA`
	`cc_idx:7 country:India`
		DBA

Risorse correlate

AWS documentazione

- AWS database, blog
- Guida alla migrazione da Oracle ad Aurora da PostgreSQL

Altra documentazione

- Array associativi Oracle
- Funzioni e operatori dell'array PostgreSQL
- Tipi PostgreSQL definiti dall'utente

Abilita connessioni crittografate per le istanze DB PostgreSQL in Amazon RDS

Rohit Kapoor, Amazon Web Services

Riepilogo

Amazon Relational Database Service (Amazon RDS) supporta la crittografia SSL per le istanze DB PostgreSQL. Utilizzando SSL, puoi crittografare una connessione PostgreSQL tra le tue applicazioni e le istanze DB di Amazon RDS for PostgreSQL. Per impostazione predefinita, Amazon RDS SSL/TLS for PostgreSQL utilizza e prevede che tutti i client si connettano utilizzando la crittografia. SSL/TLS Amazon RDS per PostgreSQL supporta le versioni TLS 1.1 e 1.2.

Questo modello descrive come abilitare le connessioni crittografate per un'istanza DB Amazon RDS for PostgreSQL. Puoi utilizzare lo stesso processo per abilitare le connessioni crittografate per Amazon Aurora PostgreSQL Compatible Edition.

Prerequisiti e limitazioni

- Un account AWS attivo
- Un'istanza DB Amazon RDS per PostgreSQL
- Un pacchetto SSL

Architecture

Tools (Strumenti)

- pgAdmin è una piattaforma di amministrazione e sviluppo open source per PostgreSQL. Puoi usare pgAdmin su Linux, Unix, macOS e Windows per gestire gli oggetti del database in PostgreSQL 10 e versioni successive.
- Gli editor di PostgreSQL forniscono un'interfaccia più intuitiva per aiutarti a creare, sviluppare ed eseguire query e modificare il codice in base alle tue esigenze.

Best practice

- Monitora le connessioni non sicure al database.

- Verifica i diritti di accesso al database.
- Assicurati che i backup e le istantanee siano crittografati quando sono inattivi.
- Monitora l'accesso al database.
- Evita i gruppi di accesso senza restrizioni.
- Migliora le tue notifiche con Amazon GuardDuty.
- Monitora regolarmente l'aderenza alle politiche.

Epiche

Scarica un certificato affidabile e importalo nel tuo negozio di fiducia

Operazione	Description	Competenze richieste
Carica un certificato affidabile sul tuo computer.	Per aggiungere certificati all'archivio Trusted Root Certification Authorities del computer, procedi nel seguente modo. (Queste istruzioni utilizzano Windows Server come esempio.) 1. In Windows Server, scegliete Start, Esegui, quindi digitate mmc. 2. Nella console, scegli File, Aggiungi/Rimuovi snap-in. 3. In Snap-in disponibili, scegli Certificati, quindi scegli Aggiungi. 4. In Questo snap-in gestirà sempre i certificati per, scegli Account computer, Avanti. 5. Scegli Computer locale, Fine.	DevOps ingegnere, ingegnere addetto alla migrazione, DBA

Operazione	Description	Competenze richieste
	6. Se non hai altri snap-in da aggiungere alla console, scegli OK. 7. Nell'albero della console, fai doppio clic su Certificati. 8. Fate clic con il pulsante destro del mouse su Trusted 9. Scegli Tutte le attività, Importa per importare i certificati scaricati. 10. Segui i passaggi della procedura guidata di importazione dei certificati.	

Forza le connessioni SSL

Operazione	Description	Competenze richieste
Creare un gruppo di parametri e impostare il parametro rds.force_ssl.	Se l'istanza DB PostgreSQL ha un gruppo di parametri personalizzato, modifica il gruppo di parametri e passa a 1. `rds.force_ssl` Se l'istanza DB utilizza il gruppo di parametri predefinito che non è `rds.force_ssl` abilitato, crea un nuovo gruppo di parametri. Puoi modificare il nuovo gruppo di parametri utilizzando l'API Amazon RDS o manualmente	DevOps ingegnere, ingegnere addetto alla migrazione, DBA

Operazione	Description	Competenze richieste
	come indicato nelle seguenti istruzioni. Per creare un nuovo gruppo di parametri: 1. Accedi alla console di gestione AWS e apri la console Amazon RDS per la regione AWS che ospita l'istanza DB. 2. Nel riquadro di navigazione scegliere Parameter groups (Gruppi di parametri). 3. Scegli Crea gruppo di parametri e imposta i seguenti valori: • Per la famiglia di gruppi di parametri, scegli postgres14. • <database_instance>Per Nome gruppo, digita pgsql - -ssl. • In Descrizione, inserisci una descrizione in formato libero per il gruppo di parametri che stai aggiungendo. • Scegli Create (Crea). 4. Scegli il gruppo di parametri che hai creato. 5. Da Parameter group actions (Operazioni gruppo di	

Operazione	Description	Competenze richieste
	parametri) scegliere Edit (Modifica). 6. Trova rds.force_ssl e modificane l'impostazione su 1. **Note** Esegui test sul lato client prima di modificare questo parametro. 7. Scegli Save changes (Salva modifiche). Per associare il gruppo di parametri alla tua istanza DB PostgreSQL: 1. Sulla console Amazon RDS, nel riquadro di navigazione, scegli Database, quindi scegli l'istanza DB PostgreSQL. 2. Scegli Modifica. 3. In Configurazione aggiuntiva, scegli il nuovo gruppo di parametri, quindi scegli Continua. 4. In Pianifica modifiche, scegli Applica immediatamente.	

Operazione	Description	Competenze richieste
	5. Scegliere Modify DB Instance (Modifica istanza database). Per ulteriori informazioni, consulta la documentazione di Amazon RDS.	
Forza le connessioni SSL.	Connettiti all'istanza DB Amazon RDS for PostgreSQL. I tentativi di connessione che non utilizzano SSL vengono rifiutati con un messaggio di errore. Per ulteriori informazioni, consulta la documentazione di Amazon RDS.	DevOps ingegnere, ingegnere addetto alla migrazione, DBA

Installa l'estensione SSL

Operazione	Description	Competenze richieste
Installa l'estensione SSL.	1. Avvia una connessione psql o pgAdmin come DBA. 2. Chiamate la funzione ssl_is_used () per determinare se viene utilizzato SSL. `select ssl_is_used();` La funzione restituisce `t` se la connessione utilizza SSL; in caso contrario, restituisce. `f` 3. Installa l'estensione SSL.	DevOps ingegnere, ingegnere addetto alla migrazione, DBA

Operazione	Description	Competenze richieste
	`create extension sslinfo; show ssl; select ssl_cipher();` Per ulteriori informazioni, consulta la documentazione di Amazon RDS.	

Configura il tuo client PostgreSQL per SSL

Operazione	Description	Competenze richieste
Configura un client per SSL.	Utilizzando SSL, è possibile avviare il server PostgreSQL con il supporto per connessioni crittografate che utilizzano i protocolli TLS. Il server ascolta le connessioni standard e SSL sulla stessa porta TCP e negozia con qualsiasi client connesso se utilizzare SSL. Per impostazione predefinita, questa è un'opzione client. Se stai usando il client psql: 1. Assicurati che il certificato Amazon RDS sia stato caricato sul tuo computer locale. 2. Avvia una connessione client SSL aggiungendo quanto segue:	DevOps ingegnere, ingegnere addetto alla migrazione, DBA

Operazione	Description	Competenze richieste

	`psql postgres -h SOMEHOST.amazonaws.com -p 8192 -U someuser sslmode=verify-full sslrootcert=rds-ssl-ca-cert.pem`
	`select ssl_cipher();`
	
	Per altri client PostgreSQL:
	
	• Modifica il rispettivo parametro della chiave pubblica dell'applicazione. Questo potrebbe essere disponibile come opzione, come parte della stringa di connessione o come proprietà nella pagina di connessione negli strumenti della GUI.
	
	Consulta le seguenti pagine per questi client:
	
	• Documentazione pgAdmin
	• Documentazione JDBC	

risoluzione dei problemi

Problema	Soluzione
Impossibile scaricare il certificato SSL.	Verifica la connessione al sito Web e riprova a scaricare il certificato sul computer locale.

Risorse correlate

- Documentazione Amazon RDS per PostgreSQL
- Utilizzo di SSL con un'istanza DB PostgreSQL (documentazione Amazon RDS)
- TCP/IP Connessioni sicure con SSL (documentazione PostgreSQL)
- Utilizzo di SSL (documentazione JDBC)

Crittografa un'istanza database Amazon RDS for PostgreSQL esistente

Piyush Goyal, Shobana Raghu e Yaser Raja, Amazon Web Services

Riepilogo

Questo modello spiega come crittografare un'istanza database Amazon Relational Database Service (Amazon RDS) per PostgreSQL esistente nel cloud AWS con tempi di inattività minimi. Questo processo funziona anche per le istanze DB di Amazon RDS for MySQL.

Puoi abilitare la crittografia per un'istanza DB Amazon RDS al momento della creazione, ma non dopo la creazione. Tuttavia, puoi aggiungere la crittografia a un'istanza DB non crittografata creando uno snapshot dell'istanza DB e quindi creando una copia crittografata di tale istantanea. È quindi possibile ripristinare un'istanza DB dallo snapshot crittografato per ottenere una copia crittografata dell'istanza DB originale. Se il progetto prevede tempi di inattività (almeno per le transazioni di scrittura) durante questa attività, questo è tutto ciò che devi fare. Quando la nuova copia crittografata dell'istanza DB diventa disponibile, puoi indirizzare le applicazioni verso il nuovo database. Tuttavia, se il progetto non prevede tempi di inattività significativi per questa attività, è necessario un approccio alternativo che consenta di ridurre al minimo i tempi di inattività. Questo modello utilizza AWS Database Migration Service (AWS DMS) per migrare e replicare continuamente i dati in modo che il passaggio al nuovo database crittografato possa essere eseguito con tempi di inattività minimi.

Le istanze DB crittografate di Amazon RDS utilizzano l'algoritmo di crittografia AES-256 standard del settore per crittografare i dati sul server che ospita le istanze database di Amazon RDS. Dopo la crittografia dei dati, Amazon RDS gestisce l'autenticazione dell'accesso e la decrittografia dei dati in modo trasparente, con un impatto minimo sulle prestazioni. Non è quindi necessario modificare le applicazioni client di database per utilizzare la crittografia.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un'istanza database Amazon RDS for PostgreSQL non crittografata
- Esperienza nell'utilizzo (creazione, modifica o interruzione) di attività AWS DMS (consulta Lavorare con le attività di AWS DMS nella documentazione di AWS DMS)
- Familiarità con AWS Key Management Service (AWS KMS) per la crittografia dei database (consulta la documentazione di AWS KMS)

Limitazioni

- Puoi abilitare la crittografia per un'istanza DB Amazon RDS solo al momento della creazione, non dopo la creazione dell'istanza DB.
- I dati nelle tabelle non registrate non verranno ripristinati utilizzando istantanee. Per ulteriori informazioni, consulta Best practice for working with PostgreSQL.
- Non è possibile creare una replica di lettura crittografata di un'istanza database non crittografata o una replica di lettura non crittografata di un'istanza database crittografata.
- Non puoi ripristinare un backup o uno snapshot non crittografato in un'istanza database crittografata.
- AWS DMS non trasferisce automaticamente le sequenze, pertanto sono necessari passaggi aggiuntivi per gestirlo.

Per ulteriori informazioni, consulta Limitazioni delle istanze DB crittografate di Amazon RDS nella documentazione di Amazon RDS.

Architecture

Architettura di origine

- Istanza DB RDS non crittografata

Architettura Target

- Istanza DB RDS crittografata
  - L'istanza DB RDS di destinazione viene creata ripristinando la copia istantanea DB dell'istanza DB RDS di origine.
  - Una chiave AWS KMS viene utilizzata per la crittografia durante il ripristino dello snapshot.
  - Un'attività di replica AWS DMS viene utilizzata per migrare i dati.

Tools (Strumenti)

Strumenti utilizzati per abilitare la crittografia:

- Chiave AWS KMS per la crittografia: quando crei un'istanza DB crittografata, puoi scegliere una chiave gestita dal cliente o la chiave gestita AWS per Amazon RDS per crittografare la tua istanza DB. Se non specifichi l'identificatore di chiave per una chiave gestita dal cliente, Amazon RDS utilizza la chiave gestita AWS per la tua nuova istanza DB. Amazon RDS crea una chiave gestita AWS per Amazon RDS per il tuo account AWS. Il tuo account AWS ha una chiave gestita AWS diversa per Amazon RDS per ogni regione AWS. Per ulteriori informazioni sull'uso delle chiavi KMS per la crittografia Amazon RDS, consulta Encrypting Amazon RDS Resources.

Strumenti utilizzati per la replica continua:

- AWS DMS: puoi utilizzare AWS Database Migration Service (AWS DMS) per replicare le modifiche dal DB di origine al DB di destinazione. È importante mantenere sincronizzati il DB di origine e quello di destinazione per ridurre al minimo i tempi di inattività. Per informazioni sulla configurazione di AWS DMS e sulla creazione di attività, consulta la documentazione di AWS DMS.

Epiche

Crea un'istantanea dell'istanza DB di origine e crittografala

Operazione	Description	Competenze richieste
Controlla i dettagli dell'istanza database PostgreSQL di origine.	Sulla console Amazon RDS, scegli l'istanza database PostgreSQL di origine. Nella scheda Configurazione, assicurati che la crittografia non sia abilitata per l'istanza . Per un'illustrazione dello schermo, consulta la sezione Informazioni aggiuntive.	DBA
Crea l'istantanea del DB.	Crea uno snapshot DB dell'istanza che desideri crittografare. Il tempo necessario per creare uno snapshot dipende dalla dimensione del database. Per istruzioni, consulta Creazione	DBA

Operazione	Description	Competenze richieste
	di uno snapshot DB nella documentazione di Amazon RDS.	
Crittografa l'istantanea.	Nel pannello di navigazione della console Amazon RDS, scegli Snapshot e seleziona lo snapshot DB che hai creato. In Actions (Operazioni), selezionare Copy Snapshot (Copia snapshot). Fornisci la regione AWS di destinazi one e il nome della copia dello snapshot del DB nei campi corrispondenti. Seleziona la casella di controllo Abilita crittografia. Per Master Key (Chiave master), specifica l'identificatore di chiave KMS da utilizzare per crittogra fare la copia di snapshot DB. Seleziona Copy Snapshot (Copia snapshot). Per ulteriori informazioni, consulta Copiare uno snapshot nella documenta zione di Amazon RDS.	DBA

Preparare l'istanza DB di destinazione

Operazione	Description	Competenze richieste
Ripristina l'istantanea del DB.	Sulla console Amazon RDS, scegli Snapshots. Scegli lo snapshot crittografato che hai creato. Per Actions (Operazio	DBA

Operazione	Description	Competenze richieste
	ni), selezionare Restore Snapshot (Ripristina snapshot) . Per DB Instance Identifie r, fornisci un nome univoco per la nuova istanza DB. Esamina i dettagli dell'istanza, quindi scegli Ripristina istanza DB. Una nuova istanza DB crittografata verrà creata dalla tua istantanea. Per ulteriori informazioni, consulta Ripristin o da uno snapshot DB nella documentazione di Amazon RDS.	
Migra i dati utilizzando AWS DMS.	Sulla console AWS DMS, crea un'attività AWS DMS. Per il tipo di migrazione, scegli Migra i dati esistenti e replica le modifiche in corso. In Impostazioni attività, per la modalità di preparazione della tabella di Target, scegli Tronca. Per ulteriori informazi oni, consulta Creazione di un'attività nella documenta zione di AWS DMS.	DBA
Abilita la convalida dei dati.	In Impostazioni attività, scegli Abilita convalida. Ciò consente di confrontare i dati di origine con i dati di destinazione per verificare che i dati siano stati migrati correttamente.	DBA

Operazione	Description	Competenze richieste
Disabilita i vincoli sull'istanza DB di destinazione.	Disabilita eventuali trigger e vincoli di chiave esterna sull'istanza DB di destinazione, quindi avvia l'attività AWS DMS. Per ulteriori informazioni sulla disabilitazione dei trigger e dei vincoli di chiave esterna, consulta la documentazione di AWS DMS.	DBA
Verifica i dati.	Al termine del caricamento completo, verifica i dati sull'istanza DB di destinazione per vedere se corrispondono ai dati di origine. Per ulteriori informazioni, consulta la convalida dei dati di AWS DMS nella documentazione di AWS DMS.	DBA

Passa all'istanza DB di destinazione

Operazione	Description	Competenze richieste
Interrompi le operazioni di scrittura sull'istanza DB di origine.	Interrompi le operazioni di scrittura sull'istanza DB di origine in modo che possa iniziare il downtime dell'applicazione. Verifica che AWS DMS abbia completato la replica dei dati nella pipeline. Abilita i trigger e le chiavi esterne sull'istanza DB di destinazione.	DBA

Operazione	Description	Competenze richieste
Aggiorna le sequenze del database	Se il database di origine contiene numeri di sequenza, verificate e aggiornate le sequenze nel database di destinazione.	DBA
Configura l'endpoint dell'appl icazione.	Configura le connessioni delle applicazioni per utilizzare i nuovi endpoint di istanze database Amazon RDS. L'istanza DB è ora crittogra fata.	DBA, proprietario dell'appl icazione

Risorse correlate

- Creazione di un task AWS DMS
- Monitoraggio delle attività di replica tramite Amazon CloudWatch
- Monitoraggio delle attività di AWS DMS
- Aggiornamento della chiave di crittografia Amazon RDS

Informazioni aggiuntive

Verifica della crittografia per l'istanza database PostgreSQL di origine:

Note aggiuntive per questo modello:

- Abilita la replica su PostgreSQL impostando il parametro su 1. `rds.logical_replication`

Nota importante: gli slot di replica conservano i file WAL (Write Ahead Log) fino a quando i file non vengono consumati esternamente, ad esempio da `pg_recvlogical` processi di estrazione, trasformazione e caricamento (ETL) o da AWS DMS. Quando imposti il valore del `rds.logical_replication` parametro su 1, AWS DMS imposta i `max_connections` parametri `wal_levelmax_wal_senders`,`max_replication_slots`, e. Se sono presenti slot di replica

logici ma non esiste alcun utente per i file WAL conservati dallo slot di replica, è possibile che si verifichi un aumento dell'utilizzo del disco del registro delle transazioni e una diminuzione costante dello spazio di archiviazione libero. Per ulteriori informazioni e passaggi per risolvere questo problema, consulta l'articolo Come posso identificare la causa dell'errore «Nessuno spazio rimasto sul dispositivo» o "DiskFull" su Amazon RDS for PostgreSQL? nel Knowledge Center di AWS Support.

- Qualsiasi modifica allo schema apportata all'istanza DB di origine dopo aver creato lo snapshot DB non sarà presente nell'istanza DB di destinazione.
- Dopo aver creato un'istanza DB crittografata, non è possibile modificare la chiave KMS utilizzata da quell'istanza DB. Assicurati di determinare i requisiti della chiave KMS prima di creare l'istanza DB crittografata.
- È necessario disabilitare i trigger e le chiavi esterne sull'istanza DB di destinazione prima di eseguire l'attività AWS DMS. Puoi riattivarli quando l'attività è completa.

Applica il tagging automatico dei database Amazon RDS al momento del lancio

Susanne Kangnoh e Archit Mathur, Amazon Web Services

Riepilogo

Amazon Relational Database Service (Amazon RDS) è un servizio Web che semplifica la configurazione, il funzionamento e la scalabilità di un database relazionale nel cloud Amazon Web Services (AWS). Offre una capacità ridimensionabile a un costo conveniente per un database relazionale standard del settore e gestisce task comuni di amministrazione del database.

Puoi utilizzare i tag per classificare le tue risorse AWS in diversi modi. L'etichettatura dei database relazionali è utile quando hai molte risorse nel tuo account e desideri identificare rapidamente una risorsa specifica in base ai tag. Puoi utilizzare i tag Amazon RDS per aggiungere metadati personalizzati alle tue istanze DB RDS. Un tag è costituito da una chiave e un valore definiti dall'utente. Ti consigliamo di creare un set coerente di tag per soddisfare i requisiti della tua organizzazione.

Questo modello fornisce un CloudFormation modello AWS per aiutarti a monitorare e contrassegnare le istanze DB RDS. Il modello crea un evento Amazon CloudWatch Events che controlla l'DBInstanceevento AWS CloudTrail Create. (CloudTrail acquisisce le chiamate API per Amazon RDS come eventi.) Quando rileva questo evento, chiama una funzione AWS Lambda che applica automaticamente le chiavi e i valori dei tag definiti dall'utente. Il modello invia anche una notifica che indica che l'istanza è stata etichettata, utilizzando Amazon Simple Notification Service (Amazon SNS).

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) per caricare il codice Lambda.
- Un indirizzo e-mail a cui desideri ricevere notifiche di tagging.

Limitazioni

- La soluzione supporta CloudTrail Create DBInstance events. Non crea notifiche per altri eventi.

Architecture

Architettura del workflow

Automazione e scalabilità

- Puoi utilizzare il CloudFormation modello AWS più volte per diverse regioni e account AWS. Devi eseguire il modello solo una volta in ogni regione o account.

Tools (Strumenti)

Servizi AWS

- AWS CloudTrail: AWS CloudTrail è un servizio AWS che ti aiuta con la governance, la conformità e il controllo operativo e del rischio del tuo account AWS. Le azioni intraprese da un utente, un ruolo o un servizio AWS vengono registrate come eventi in CloudTrail.
- Amazon CloudWatch Events: Amazon CloudWatch Events offre un flusso quasi in tempo reale di eventi di sistema che descrivono i cambiamenti nelle risorse AWS. CloudWatch Events viene a conoscenza dei cambiamenti operativi man mano che si verificano e intraprende le azioni correttive necessarie, inviando messaggi per rispondere all'ambiente, attivando funzioni, apportando modifiche e acquisendo informazioni sullo stato.
- AWS Lambda: AWS Lambda è un servizio di elaborazione che supporta l'esecuzione di codice senza la necessità di fornire o gestire server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo. Verrà addebitato soltanto il tempo di calcolo consumato e non verrà addebitato alcun costo quando il codice non è in esecuzione.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti altamente scalabile che può essere utilizzato per un'ampia gamma di soluzioni di storage, tra cui siti Web, applicazioni mobili, backup e data lake.
- Amazon SNS — Amazon Simple Notification Service (Amazon SNS) è un servizio Web che consente ad applicazioni, utenti finali e dispositivi di inviare e ricevere istantaneamente notifiche dal cloud.

Codice

Questo modello include un allegato con due file:

- `index.zip`è un file compresso che include il codice Lambda per questo modello.
- `rds.yaml`è un CloudFormation modello che distribuisce il codice Lambda.

Consulta la sezione Epics per informazioni su come usare questi file.

Epiche

Implementa il codice Lambda

Operazione	Description	Competenze richieste
Carica il codice in un bucket S3.	Crea un nuovo bucket S3 o usa un bucket S3 esistente per caricare il file allegato `index.zip` (codice Lambda). Questo bucket deve trovarsi nella stessa regione AWS delle risorse (istanze DB RDS) che desideri monitorare.	Architetto del cloud
Implementa il CloudFormation modello.	Apri la console Cloudformation nella stessa regione AWS del bucket S3 e distribuisci il `rds.yaml` file fornito nell'alle gato. Nella prossima epopea, fornisci i valori per i parametri del modello.	Architetto del cloud

Completa i parametri nel CloudFormation modello

Operazione	Description	Competenze richieste
Fornisci il nome del bucket S3.	Inserisci il nome del bucket S3 che hai creato o seleziona to nella prima epic. Questo bucket S3 contiene il file.zip	Architetto del cloud

Operazione	Description	Competenze richieste
	per il codice Lambda e deve trovarsi nella stessa regione AWS del CloudFormation modello e delle istanze DB RDS che desideri monitorare.	
Fornisci la chiave S3.	Fornisci la posizione del file.zip del codice Lambda nel tuo bucket S3, senza barre iniziali (ad esempio o). `index.zip` `controls/index.zip`	Architetto del cloud
Fornisci un indirizzo email.	Fornisci un indirizzo email attivo a cui desideri ricevere le notifiche di violazione.	Architetto del cloud
Specificare un livello di registrazione.	Specificare il livello di registrazione e la verbosità. `Info`indica messaggi informativi dettagliati sullo stato di avanzamento dell'applicazione e deve essere utilizzato solo per il debug. `Error`indica eventi di errore che potrebbero comunque consentire all'applicazione di continuare a funzionare. `Warning`indica situazioni potenzialmente dannose.	Architetto del cloud

Operazione	Description	Competenze richieste
Inserisci le chiavi e i valori dei tag per le tue istanze DB RDS.	Inserisci le chiavi e i valori dei tag richiesti che desideri applicare automaticamente all'istanza RDS. Per ulteriori informazioni, consulta Tagging delle risorse Amazon RDS nella documentazione AWS.	Architetto del cloud

Confermare la sottoscrizione

Operazione	Description	Competenze richieste
Conferma l'iscrizione via e-mail.	Quando il CloudFormation modello viene distribuito correttamente, invia un messaggio e-mail di sottoscrizione all'indirizzo e-mail fornito. Per ricevere notifiche quando le istanze vengono contrassegnate, è necessario confermare questa sottoscrizione e-mail.	Architetto del cloud

Risorse correlate

- Creazione di un bucket (documentazione Amazon S3)
- Etichettatura delle risorse Amazon RDS (documentazione Amazon Aurora)
- Caricamento di oggetti (documentazione Amazon S3)
- Creazione di una regola CloudWatch Events che si attiva su una chiamata API AWS utilizzando AWS CloudTrail (CloudWatch documentazione Amazon)

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Stima del costo di una tabella DynamoDB per la capacità su richiesta

Moinul Al-Mamun, Amazon Web Services

Riepilogo

Amazon DynamoDB è un database transazionale NoSQL che fornisce una latenza di un millisecondo anche su scala di petabyte. Questa offerta serverless di Amazon Web Services (AWS) sta diventando popolare grazie alle sue prestazioni e scalabilità costanti. Non è necessario effettuare il provisioning dell'infrastruttura sottostante. La singola tabella può crescere fino a petabyte.

Con la modalità di capacità su richiesta, paghi in base alla richiesta per le letture e le scritture dei dati eseguite dall'applicazione sulle tabelle. I costi di AWS si basano sulle unità di richiesta di lettura (RRUs) e unità di richiesta di scrittura (WRUs) accumulate in un mese. DynamoDB monitora continuamente le dimensioni della tabella durante tutto il mese per determinare i costi di archiviazione. Supporta il backup continuo con point-in-time-recovery (PITR). DynamoDB monitora continuamente le dimensioni delle tabelle abilitate a PITR per tutto il mese per determinare i costi di backup.

Per stimare il costo di DynamoDB per un progetto, è importante calcolare la quantità di RRU, WRU e storage che verrà consumata nelle diverse fasi del ciclo di vita del prodotto. Per una stima approssimativa dei costi, puoi utilizzare AWS Pricing Calculator, ma devi fornire un numero approssimativo di RRUs requisiti di storage per la tua tabella. WRUs Questi possono essere difficili da stimare all'inizio del progetto. AWS Pricing Calculator non considera il tasso di crescita dei dati o la dimensione degli articoli e non considera separatamente il numero di letture e scritture per la tabella di base e gli indici secondari globali (). GSIs Per utilizzare AWS Pricing Calculator, devi stimare tutti questi aspetti e ipotizzare cifre approssimative per WRU, RRU e dimensioni dello storage per ottenere una stima dei costi.

Questo modello fornisce un meccanismo e un modello Microsoft Excel riutilizzabile per stimare i fattori di costo di base di DynamoDB, come i costi di scrittura, lettura, archiviazione, backup e ripristino, per la modalità di capacità on demand. È più granulare di AWS Pricing Calculator e considera la tabella di base e GSIs i requisiti in modo indipendente. Considera inoltre il tasso di crescita mensile dei dati relativi agli articoli e prevede i costi per tre anni.

Prerequisiti e limitazioni

Prerequisiti

- Conoscenza di base di DynamoDB e della progettazione di modelli di dati DynamoDB

- Conoscenze di base sui prezzi di DynamoDB, WRU, RRU, storage, backup e ripristino (per ulteriori informazioni, consulta Prezzi per la capacità on demand)
- Conoscenza dei dati, del modello di dati e delle dimensioni degli elementi in DynamoDB
- Conoscenza di DynamoDB GSIs

Limitazioni

- Il modello fornisce un calcolo approssimativo, ma non è appropriato per tutte le configurazioni. Per ottenere una stima più accurata, è necessario misurare la dimensione del singolo articolo per ogni articolo nella tabella di base e. GSIs
- Per una stima più accurata, è necessario considerare il numero previsto di scritture (inserimento, aggiornamento ed eliminazione) e letture per ogni articolo in un mese medio.
- Questo modello supporta la stima dei soli costi di scrittura, lettura, archiviazione, backup e ripristino per i prossimi anni sulla base di ipotesi di crescita fissa dei dati.

Tools (Strumenti)

Servizi AWS

- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.

Altri strumenti

- AWS Pricing Calculator è uno strumento di pianificazione basato sul Web che puoi utilizzare per creare stime per i tuoi casi d'uso AWS.

Best practice

Per mantenere bassi i costi, prendi in considerazione le seguenti best practice di progettazione di DynamoDB.

- Progettazione delle chiavi di partizione: utilizza una chiave di partizione ad alta cardinalità per distribuire il carico in modo uniforme.
- Modello di progettazione dell'elenco di adiacenza: utilizza questo modello di progettazione per la gestione one-to-many e le relazioni. many-to-many

- Indice sparso: usa l'indice sparso per il tuo. GSIs Quando crei un GSI, specifica una chiave di partizione e opzionalmente una chiave di ordinamento. Solo gli elementi della tabella di base che contengono una chiave di partizione GSI corrispondente vengono visualizzati nell'indice sparse. Questo aiuta a rimanere GSIs più piccoli.
- Sovraccarico dell'indice: utilizza lo stesso GSI per indicizzare vari tipi di articoli.
- Partizionamento in scrittura del GSI: partiziona con cura per distribuire i dati tra le partizioni per query efficienti e veloci.
- Oggetti di grandi dimensioni: archivia solo i metadati all'interno della tabella, salva il blob in Amazon S3 e conserva il riferimento in DynamoDB. Suddividi gli elementi di grandi dimensioni in più elementi e indicizzali in modo efficiente utilizzando le chiavi di ordinamento.

Per altre best practice di progettazione, consulta la Guida per gli sviluppatori di Amazon DynamoDB.

Epiche

Estrai le informazioni sugli elementi dal tuo modello di dati DynamoDB

Operazione	Description	Competenze richieste
Ottieni le dimensioni dell'arti colo.	1. Controlla quanti tipi diversi di articoli riporrai nella tua tabella. 2. Per calcolare la dimensione di ogni elemento in kilobyte, aggiungi le dimensioni della chiave e del valore di ciascun attributo. 3. Calcola la dimensione dell'elemento per una tabella di base e per ogni GSI.	Ingegnere dei dati
Stima il costo di scrittura.	Per stimare i costi di scrittura in modalità di capacità on demand, devi innanzitutto misurare quanti ne WRUs	Ingegnere dei dati

Operazione	Description	Competenze richieste

	verranno consumati in un mese. A tal fine, è necessario considerare i seguenti fattori:
	• Numero di operazioni di creazione, aggiornamento ed eliminazione per ogni elemento in un mese.
	• Numero di opzioni disponibili GSIs. Considera ogni indice in modo indipendente.
	  • Dimensione media di un elemento dell'indice
	  • Numero di tempi di sincronizzazione su un indice
	• Quanti nuovi elementi (ad esempio componenti o prodotti) verranno aggiunti alla tabella ogni mese? Il numero di elementi aggiunti potrebbe variare ogni mese, ma puoi ipotizzare un tasso di crescita medio in base ai tuoi casi aziendali.
	
	Per ulteriori informazioni, consulta la sezione Informazioni aggiuntive.	

Operazione	Description	Competenze richieste
Stima il costo di lettura.	Per stimare i costi di lettura in modalità on demand, devi innanzitutto misurare quanti ne RRUs verranno consumati in un mese. A tal fine, è necessario considerare i seguenti fattori: • Numero di disponibili GSIs. Considera ogni indice in modo indipendente. • Dimensione media di un elemento dell'indice • Numero medio di letture per prodotto al mese. • Numero totale di elementi disponibili (componenti o prodotti) nella tabella DynamoDB.	Ingegnere dei dati, sviluppatore di app

Operazione	Description	Competenze richieste
Stima le dimensioni e i costi di archiviazione.	Innanzitutto, stima il fabbisogno medio mensile di archiviazione in base alle dimensioni dell'articolo nella tabella. Quindi calcola il costo di storage moltiplicando la dimensione dello storage per il prezzo di storage per GB per la tua regione AWS. Se hai già inserito dati per stimare il costo di scrittura, non è necessario inserirli nuovamente per calcolare le dimensioni dello storage. Altrimenti, per stimare le dimensioni dello spazio di archiviazione, è necessario considerare i seguenti fattori: • Numero di elementi di dati in un modulo (prodotto) in base al design della tabella. • Dimensione media degli articoli in kilobyte. • Numero di articoli disponibili. GSIs Considera ogni indice in modo indipendente. • Dimensione media di un elemento dell'indice • Quanti nuovi prodotti verranno aggiunti alla tabella ogni mese? Il	Ingegnere dei dati

Operazione	Description	Competenze richieste
	numero di nuovi prodotti potrebbe variare ogni mese, ma puoi ipotizzare un tasso di crescita medio in base ai tuoi casi aziendali. Questo esempio utilizza una media di 10 milioni di nuovi prodotti ogni mese.	

Inserisci le informazioni sull'articolo e sull'oggetto nel modello Excel

Operazione	Description	Competenze richieste
Scarica il modello Excel dalla sezione Allegati e adattalo alla tabella dei casi d'uso.	1. Scarica il modello Excel. 2. Modifica il modulo aziendale e GSIs, in base al design del tuo tavolo.	Ingegnere dei dati
Inserisci le informazioni nel modello Excel.	1. Aggiorna le informazioni sull'articolo nel foglio. Aggiorna i dati solo nelle celle arancioni. 2. Modifica i numeri degli oggetti: quanto potrebbe essere aggiunto alla tabella ogni mese? 3. Aggiorna i prezzi per milione di WRU e RRU per la tua regione AWS. 4. Aggiorna i prezzi di storage e backup per GB al mese per la tua regione AWS.	Ingegnere dei dati

Operazione	Description	Competenze richieste
	5. Aggiorna il prezzo di ripristino per GB per la tua regione AWS. Nel modello sono presenti tre elementi o entità: informazi oni, metadati e relazioni. Ce ne sono due GSIs. Per il tuo caso d'uso, se hai bisogno di più elementi, crea nuove righe. Se ne hai bisogno di più GSIs, copia un blocco GSI esistente e incollalo per creare tutti i blocchi GSI di cui hai bisogno. Quindi regola i calcoli delle colonne SUM e TOTAL.	

Risorse correlate

Riferimenti

- Prezzi di Amazon DynamoDB per la capacità su richiesta
- Calcolatore dei prezzi AWS per DynamoDB
- Best practice per la progettazione e l'architettura con DynamoDB
- Nozioni di base su DynamoDB

Guide e pattern

- Modellazione dei dati con Amazon DynamoDB
- Stima dei costi di storage per una tabella Amazon DynamoDB

Informazioni aggiuntive

Scrivi un esempio di calcolo dei costi

Il design del modello di dati DynamoDB mostra tre elementi per prodotto e una dimensione media degli elementi di 4 KB. Quando aggiungi un nuovo prodotto alla tabella base di DynamoDB, consuma il numero di elementi* (dimensione dell'articolo/1 KB di unità di scrittura) = 3 * (4/1) = 12 WRU. In questo esempio, per scrivere 1 KB, il prodotto consuma 1 WRU.

Leggi l'esempio di calcolo dei costi

Per ottenere la stima della RRU, considera la media di quante volte ogni articolo verrà letto in un mese. Ad esempio, l'elemento Informazioni verrà letto, in media, 10 volte in un mese, l'elemento di metadati verrà letto due volte e l'elemento relativo alla relazione verrà letto cinque volte. Nel modello di esempio, RRU totale per tutti i componenti = numero di nuovi componenti creati ogni mese * RRU per componente al mese = 10 milioni * 17 RRU = 170 milioni di RRU al mese.

Ogni mese verranno aggiunti nuovi elementi (componenti o prodotti) e il numero totale di prodotti aumenterà nel tempo. Pertanto, anche i requisiti RRU aumenteranno nel tempo.

- Per il primo mese RRU, il consumo sarà di 170 milioni.
- Per il secondo mese, il consumo di RRU sarà di 2* 170 milioni = 340 milioni.
- Per il terzo mese il consumo di RRU sarà di 3* 170 milioni = 510 milioni.

Il grafico seguente mostra il consumo mensile di RRU e le previsioni dei costi.

Nota che i prezzi nel grafico sono solo a scopo illustrativo. Per creare previsioni accurate per il tuo caso d'uso, consulta la pagina dei prezzi di AWS e utilizza i prezzi nel foglio Excel.

Esempi di calcolo dei costi di storage, backup e ripristino

Lo storage, il backup e il ripristino di DynamoDB sono tutti collegati tra loro. Il backup è direttamente collegato allo storage e il ripristino è direttamente collegato alle dimensioni del backup. All'aumentare delle dimensioni della tabella, i costi di archiviazione, backup e ripristino corrispondenti aumenteranno proporzionalmente.

Dimensioni e costi dello storage

Il costo dello storage aumenterà nel tempo in base al tasso di crescita dei dati. Ad esempio, supponiamo che la dimensione media di un componente o prodotto nella tabella di base GSIs sia di 11 KB e che 10 milioni di nuovi prodotti vengano aggiunti ogni mese alla tabella del database. In tal caso, la dimensione della tabella DynamoDB aumenterà (11 KB * 10 milioni) /1024/1024 = 105 GB

al mese. Nel primo mese, la dimensione di archiviazione della tabella sarà di 105 GB, nel secondo mese sarà di 105 + 105 = 210 e così via. GBs

- Per il primo mese, il costo di storage sarà di 105 GB* al prezzo di storage per GB per la tua regione AWS.
- Per il secondo mese, il costo di storage sarà di 210 GB* al prezzo di archiviazione per GB per ogni regione.
- Per il terzo mese, il costo di archiviazione sarà di 315 GB*, il prezzo di archiviazione per GB per regione.

Per le dimensioni e i costi dello storage per i prossimi tre anni, consulta la sezione Dimensioni e previsioni dello storage.

Costo di backup

I costi di backup aumenteranno nel tempo in base al tasso di crescita dei dati. Quando si attiva il backup continuo con point-in-time-recovery (PITR), i costi di backup continuo si basano sulla media di storage in GB al mese. In un mese solare, la dimensione media del backup sarebbe la stessa della dimensione di archiviazione della tabella, anche se le dimensioni effettive potrebbero essere leggermente diverse. Man mano che verranno aggiunti nuovi prodotti ogni mese, la dimensione totale dello storage e la dimensione del backup aumenteranno nel tempo. Ad esempio, per il primo mese, la dimensione media di backup di 105 GB potrebbe aumentare fino a 210 GB per il secondo mese.

- Per il primo mese, il costo del backup sarà di 105 GB* al mese (prezzo per GB di backup continuo) per GB nella tua regione AWS.
- Per il secondo mese, il costo del backup sarà di 210 GB* al mese (prezzo per GB) di backup continuo per ogni regione.
- Per il terzo mese, il costo del backup sarà di 315 GB al mese*, il prezzo per GB del backup continuo a seconda della regione.
- e così via

I costi di Backup sono inclusi nel grafico della sezione Dimensioni dello storage e previsione dei costi.

Costo di ripristino

Quando si esegue un backup continuo con PITR abilitato, i costi delle operazioni di ripristino si basano sulle dimensioni del ripristino. Ogni volta che si esegue il ripristino, il pagamento viene

calcolato in base ai gigabyte di dati ripristinati. Se le dimensioni della tabella sono grandi e il ripristino viene eseguito più volte in un mese, l'operazione risulterà costosa.

Per stimare i costi di ripristino, questo esempio presuppone che si esegua un ripristino PITR una volta al mese alla fine del mese. L'esempio utilizza la dimensione media mensile del backup come dimensione dei dati di ripristino per quel mese. Per il primo mese, la dimensione media del backup è di 105 GB, mentre per il ripristino alla fine del mese, la dimensione dei dati di ripristino sarebbe di 105 GB. Per il secondo mese, sarebbero 210 GBs e così via.

I costi di ripristino aumenteranno nel tempo in base al tasso di crescita dei dati.

- Per il primo mese, il costo di ripristino sarà di 105 GB* al prezzo di ripristino per GB per la tua regione AWS.
- Per il secondo mese, il costo di ripristino sarà di 210 GB* al prezzo di ripristino per GB per regione.
- Per il terzo mese, il costo di ripristino sarà di 315 GB* al prezzo di ripristino per GB nella regione.

Per ulteriori informazioni, consulta la scheda Archiviazione, backup e ripristino nel modello Excel e il grafico nella sezione seguente.

Dimensioni dello storage e previsione dei costi

Nel modello, la dimensione effettiva dello spazio di archiviazione fatturabile viene calcolata sottraendo i 25 GB mensili del piano gratuito per la classe di tabelle Standard. Nel foglio, vedrai un grafico di previsione suddiviso in valori mensili.

Il seguente grafico di esempio prevede le dimensioni mensili dello storage in GB, i costi di storage fatturabili, i costi di backup su richiesta e i costi di ripristino per i prossimi 36 mesi di calendario. Tutti i costi sono in USD. Dal grafico, risulta chiaro che i costi di storage, backup e ripristino aumentano proporzionalmente all'aumento delle dimensioni dello storage.

Tieni presente che i prezzi utilizzati nel grafico sono solo a scopo illustrativo. Per creare prezzi accurati per il tuo caso d'uso, consulta la pagina dei prezzi di AWS e utilizza tali prezzi nel modello Excel.

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Stima dei costi di storage per una tabella Amazon DynamoDB

Moinul Al-Mamun, Amazon Web Services

Riepilogo

Amazon DynamoDB è un database transazionale NoSQL che fornisce una latenza di un millisecondo anche su scala di petabyte. Questa popolare offerta AWS serverless di offre prestazioni e scalabilità costanti. Non è necessario effettuare il provisioning dello storage e la singola tabella può crescere fino a petabyte.

DynamoDB monitora continuamente le dimensioni della tabella durante tutto il mese per determinare i costi di archiviazione. AWS quindi ti addebita la dimensione media dello spazio di archiviazione in gigabyte. Più la tabella cresce nel tempo, più aumenteranno i costi di archiviazione. Per calcolare i costi di archiviazione, puoi utilizzare AWS Pricing Calculator, ma devi fornire la dimensione approssimativa della tabella, compresi gli indici secondari globali (GSIs), che è davvero difficile da stimare all'inizio del progetto. Inoltre, AWS Pricing Calculator non considera il tasso di crescita dei dati.

Questo modello fornisce un meccanismo e un modello Microsoft Excel riutilizzabile per calcolare le dimensioni e i costi dello storage DynamoDB. Considera i requisiti di archiviazione per la tabella di base e quelli indipendenti. GSIs Calcola le dimensioni di archiviazione considerando le dimensioni dei singoli elementi e il tasso di crescita dei dati nel tempo.

Per ottenere una stima, inserisci due informazioni nel modello:

- La dimensione del singolo elemento in kilobyte per la tabella di base e GSIs
- Quanti nuovi oggetti o prodotti potrebbero essere aggiunti alla tabella, in media, in un mese (ad esempio, 10 milioni)

Il modello genera un grafico di archiviazione e previsione dei costi per i prossimi tre anni, come illustrato nell'esempio seguente.

Prerequisiti e limitazioni

Prerequisiti

- Conoscenza di base di DynamoDB, inclusi lo storage e i prezzi di DynamoDB

- Conoscenza dei dati, del modello di dati e delle dimensioni degli elementi in DynamoDB
- Conoscenza degli indici secondari globali DynamoDB () GSIs

Limitazioni

- Il modello fornisce un calcolo approssimativo, ma non è appropriato per tutte le configurazioni. Per ottenere una stima più accurata, è necessario misurare la dimensione del singolo articolo per ogni articolo nella tabella di base e. GSIs
- Questo modello supporta solo la stima delle dimensioni e dei costi dello storage per i prossimi anni sulla base di ipotesi di crescita fissa dei dati.

Tools (Strumenti)

Servizi AWS

- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.

Altri strumenti

- AWS Pricing Calculator è uno strumento di pianificazione basato sul Web che puoi utilizzare per creare stime per i tuoi casi AWS d'uso.

Epiche

Estrai le informazioni sugli elementi dal tuo modello di dati DynamoDB

Operazione	Description	Competenze richieste
Ottieni le dimensioni dell'arti colo.	1. Determina quanti tipi di articoli diversi vuoi memorizzare nella tua tabella. 2. Per calcolare la dimension e di ogni elemento in kilobyte, aggiungi la `Value`	Ingegnere dei dati

Operazione	Description	Competenze richieste
	dimensione Key e di ogni attributo. 3. Calcola la dimensione dell'elemento per una tabella di base e per ogni GSI.	
Ottieni il numero di oggetti aggiunti in un mese.	Stima quanti componenti o oggetti verranno aggiunti alla tabella DynamoDB, in media, in un mese.	Ingegnere dei dati

Inserisci le informazioni sull'articolo e sull'oggetto nel modello Excel

Operazione	Description	Competenze richieste
Scarica e modifica il foglio di calcolo Excel.	1. Scarica il modello Excel dal documento allegato. 2. Modifica il modulo aziendale e GSIs, in base al design della tua tabella.	Ingegnere dei dati
Inserisci le informazioni nel modello Excel.	1. Aggiorna le informazioni sull'articolo nel foglio. 2. Modifica i numeri degli oggetti: quanto potrebbe essere aggiunto alla tabella ogni mese? 3. Aggiorna il prezzo di archiviazione per GB al mese per il tuo. Regione AWS	Ingegnere dei dati

Risorse correlate

- Prezzi della capacità on demand per Amazon DynamoDB
- AWS Calcolatore dei prezzi per DynamoDB

Informazioni aggiuntive

Tieni presente che il modello allegato prevede solo le dimensioni e i costi di archiviazione per la classe di tabella di archiviazione Standard. In base alla previsione dei costi di archiviazione e considerando le dimensioni dei singoli articoli e il tasso di crescita del prodotto o dell'oggetto, è possibile stimare quanto segue:

- Costo di esportazione dei dati
- Costi di backup e ripristino
- Requisiti di archiviazione dei dati.

Costo dello storage dei dati di Amazon DynamoDB

DynamoDB monitora continuamente le dimensioni delle tabelle per determinare i costi di archiviazione. DynamoDB misura la dimensione dei dati fatturabili aggiungendo la dimensione in byte dei dati grezzi e un sovraccarico di archiviazione per elemento che dipende dalle funzionalità abilitate. Per ulteriori informazioni, consulta la DynamoDB Developer Guide.

Il prezzo per l'archiviazione dei dati dipende dalla classe di tabella. I primi 25 GB archiviati ogni mese sono gratuiti se utilizzi la classe di tabelle DynamoDB Standard. Per ulteriori informazioni sui costi di storage per le diverse classi di tabelle Standard e Standard-Infrequent Access Regioni AWS, vedi Prezzi per la capacità on demand.

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Stima le dimensioni del motore Amazon RDS per un database Oracle utilizzando i report AWR

Abhishek Verma e Eduardo Valentim, Amazon Web Services

Riepilogo

Quando esegui la migrazione di un database Oracle ad Amazon Relational Database Service (Amazon RDS) o Amazon Aurora, il calcolo della CPU, della memoria e del I/O disco per il database di destinazione è un requisito fondamentale. È possibile stimare la capacità richiesta del database di destinazione analizzando i report di Oracle Automatic Workload Repository (AWR). Questo modello spiega come utilizzare i report AWR per stimare questi valori.

Il database Oracle di origine potrebbe essere locale o ospitato su un'istanza Amazon Elastic Compute Cloud (Amazon EC2) oppure potrebbe essere un'istanza DB Amazon RDS for Oracle. Il database di destinazione potrebbe essere qualsiasi database Amazon RDS o Aurora.

> **Note**
>
> Le stime della capacità saranno più precise se il motore di database di destinazione è Oracle. Per altri database Amazon RDS, le dimensioni del motore possono variare a causa delle differenze nell'architettura del database.

Ti consigliamo di eseguire il test delle prestazioni prima di migrare il database Oracle.

Prerequisiti e limitazioni

Prerequisiti

- Una licenza Oracle Database Enterprise Edition e una licenza Oracle Diagnostics Pack per scaricare i report AWR.

Versioni del prodotto

- Tutte le edizioni di Oracle Database per le versioni 11g (versioni 11.2.0.3.v1 e successive) e fino a 12.2 e 18c,19c.
- Questo modello non copre Oracle Engineered Systems o Oracle Cloud Infrastructure (OCI).

Architecture

Stack tecnologico di origine

Una delle seguenti:

- Un database Oracle locale
- Un database Oracle su un'istanza EC2
- Un'istanza DB Amazon RDS per Oracle

Stack tecnologico Target

- Qualsiasi database Amazon RDS o Amazon Aurora

Architettura Target

Per informazioni sul processo di migrazione completo, consulta lo schema Migrare un database Oracle su Aurora PostgreSQL utilizzando AWS DMS e AWS SCT.

Automazione e scalabilità

Se hai più database Oracle da migrare e desideri utilizzare parametri prestazionali aggiuntivi, puoi automatizzare il processo seguendo i passaggi descritti nel post del blog Istanze Amazon RDS di dimensioni corrette su larga scala in base ai parametri delle prestazioni Oracle.

Tools (Strumenti)

- Oracle Automatic Workload Repository (AWR) è un repository integrato nei database Oracle. Periodicamente raccoglie e archivia i dati sull'attività del sistema e sul carico di lavoro, che vengono poi analizzati da Automatic Database Diagnostic Monitor (ADDM). AWR acquisisce istantanee dei dati sulle prestazioni del sistema periodicamente (per impostazione predefinita, ogni 60 minuti) e archivia le informazioni (per impostazione predefinita, fino a 8 giorni). È possibile utilizzare le viste e i report AWR per analizzare questi dati.

Best practice

- Per calcolare il fabbisogno di risorse per il database di destinazione, puoi utilizzare un singolo report AWR, più report AWR o viste AWR dinamiche. Si consiglia di utilizzare più report AWR

durante il periodo di picco di carico per stimare le risorse necessarie per gestire tali carichi di picco. Inoltre, le viste dinamiche forniscono più punti dati che consentono di calcolare i requisiti di risorse in modo più preciso.

- È necessario stimare gli IOPS solo per il database che si intende migrare, non per altri database e processi che utilizzano il disco.
- Per calcolare quanto I/O viene utilizzato dal database, non utilizzate le informazioni nella sezione Load Profile del rapporto AWR. Utilizza invece la sezione I/O Profilo, se disponibile, oppure vai alla sezione Instance Activity Stats e guarda i valori totali per le operazioni fisiche di lettura e scrittura.
- Quando stimi l'utilizzo della CPU, ti consigliamo di utilizzare il metodo delle metriche del database anziché le statistiche del sistema operativo (OS), perché si basa sulla CPU utilizzata solo dai database. (Le statistiche del sistema operativo includono anche l'utilizzo della CPU da parte di altri processi). È inoltre necessario controllare i consigli relativi alla CPU nel rapporto ADDM per migliorare le prestazioni dopo la migrazione.
- Quando stabilisci il tipo di istanza giusto, considera i limiti di I/O throughput (Amazon Elastic Block Store (Amazon EBS) e il throughput di rete per la dimensione specifica dell'istanza.
- Esegui il test delle prestazioni prima della migrazione per convalidare le dimensioni del motore.

Epiche

Crea un rapporto AWR

Operazione	Description	Competenze richieste
Abilita il rapporto AWR.	Per abilitare il rapporto, segui le istruzioni nella documentazione Oracle.	DBA
Verifica il periodo di conservazione.	Per verificare il periodo di conservazione del rapporto AWR, utilizza la seguente query. `SQL> SELECT snap_interval,retention FROM dba_hist_wr_control;`	DBA

Operazione	Description	Competenze richieste
Genera l'istantanea.	Se l'intervallo delle istantanee AWR non è sufficientemente granulare per registrare il picco del carico di lavoro di picco, puoi generare il rapporto AWR manualmen te. Per generare l'istantanea AWR manuale, utilizzate la seguente query. `SQL> EXEC dbms_work load_repository.cr eate_snapshot;`	DBA
Controlla le istantanee recenti.	Per controllare le istantanee AWR recenti, usa la seguente query. `SQL> SELECT snap_id, to_char(begin_inte rval_time,'dd/MON/ yy hh24:mi') Begin_Int erval, to_char(end_interv al_time,'dd/MON/yy hh24:mi') End_Interval FROM dba_hist_snapshot ORDER BY 1;`	DBA

Stima dei requisiti del disco I/O

Operazione	Description	Competenze richieste
Scegliere un metodo.	IOPS è la misura standard delle operazioni di input e	DBA

Operazione	Description	Competenze richieste
	output al secondo su un dispositivo di storage e include operazioni di lettura e scrittura . Se stai migrando un database locale in AWS, devi determina re il disco di picco I/O utilizzat o dal database. Puoi utilizzare i seguenti metodi per stimare il disco I/O per il tuo database di destinazione: • Sezione Load Profile del rapporto AWR • Sezione Instance Activity Stats del rapporto AWR (utilizzare questa sezione per Oracle Database 12c o versione successiva) • Sezione I/O Profile del rapporto AWR (utilizza re questa sezione per le versioni di Oracle Database precedenti alla 12c) • Visualizzazioni AWR I passaggi seguenti descrivon o questi quattro metodi.	

Operazione	Description	Competenze richieste
Opzione 1: utilizzare il profilo di carico.	La tabella seguente mostra un esempio della sezione Load Profile del rapporto AWR. **Important** Per informazioni più accurate, si consiglia di utilizzare l'opzione 2 (profili I/O) o l'opzione 3 (statistiche sull'attività delle istanze) anziché il profilo di carico. Al secondo / Per transazione / Per dirige / Per chiama Orari (i) DB: 26,6 / 0.2 / 0,00 / 0,02 CPU (e) DB: 18,0 / 0.1 / 0,00 / 0.01 CPU (e) in backd: 0.2 / 0,0 / 0,00 / 0,00	DBA

Operazione	Description		Competenze richieste
	Dime e di ripris o (byte	2.45 ,9	17.0
	Lettu logic (bloc :	3.37 ,5	2344
	Bloc le modi :	21.6	150,5
	Lettu fisica (bloc :	13.5	94,4
	Scrit fisica (bloc :	3.46	24,1
	Legg le richie IO:	3.58	24,9
	Scriv richie IO:	574,7	4.0

Operazione	Description		Competenze richieste
	Legg IO (MB)	106,' 0.7	
	Scriv IO (MB)	27,1 0.2	
	Righ di scan IM:	0,0 0,0	
	Sess logic Read IM:		
	Chia utent	1.24! 8.7	
	Anal (SQL	4.62(32,2	
	Anal rigide (SQL	8.9 0.1	
	Area di lavor SQL (MB)	824,! 5.7	
	Acce	1,7 0,0	

Operazione	Description	Competenze richieste
	Eseg 136.(950, (SQL Rollb 22.9 0.2 Tran 143,(ni: In base a queste informazioni, è possibile calcolare IOPs e produrre nel modo seguente: IOPS = Richieste di lettura: + I/O Richieste di scrittura = I/O 3.586,8 + 574,7 = 4134,5 Throughput = lettura fisica (blocchi) + Scrittura fisica (blocchi) = 13.575,1 + 3.467,3 = 17.042,4 Poiché la dimensione del blocco in Oracle è di 8 KB, puoi calcolare la velocità effettiva totale come segue: La velocità effettiva totale in MB è 17042,4 * 8 * 1024/1024 /1024 = 133,2 MB ⚠ Warning Non utilizzate il profilo di carico per stimare la dimension e dell'istanza. Non	

Operazione	Description	Competenze richieste
	è così preciso come le statistiche o i I/O profili delle attività delle istanze.	

Operazione	Description	Competenze richieste
Opzione 2: utilizza le statistiche sull'attività delle istanze.	Se utilizzi una versione del database Oracle precedente alla 12c, puoi utilizzare la sezione Instance Activity Stats del rapporto AWR per stimare gli IOPS e il throughput. La tabella seguente mostra un esempio di questa sezione. Statist / Totale / al secon / per Trans richies I/O totali di lettura fisica / 2.547. .217 / 3.610, / 25,11 byte totali di lettura fisica / 80.776 6.124. / 114.48 26,26 / 796.149 8 richies I/O totali di scrittu fisica / 534.19 08 / 757,11 / 5,27 byte totali di / 25.517 8.849. / 36,165 1,84 / 251,508 8	DBA

Operazione	Description	Competenze richieste
	scrittu fisici Sulla base di queste informazioni, è possibile calcolare gli IOPS totali e il throughput nel modo seguente: IOPS totali = 3.610,28 + 757,11 = 4367 Mbps totali = 114.482.426,26 + 36.165.631,84 = 150648058,1/1024/1024 = 143 Mbps	

Operazione	Description	Competenze richieste
Opzione 3: usa I/O i profili.	In Oracle Database 12c, il rapporto AWR include una sezione I/O Profili che presenta tutte le informazioni in un'unica tabella e fornisce dati più accurati sulle prestazioni del database. La tabella seguente mostra un esempio di questa sezione. Lettura +s crittura al secon Richie totali: 4.367, Richie al databa 4.161. Richie ottimiz te: 0,0 Richie di ripristi o: 179,3 Totale (MB): 143,7 Lettura al secon 3.610, 3.586, 0,0 2.8 109,2 Scrivi al secondo 757,1 574,7 0,0 176,6 34,5	DBA

Operazione	Description	Competenze richieste
	Banca dati (MB): 133,1 106,1 27,1 Totale ottimizzato (MB): 0,0 0,0 0,0 Ripeti (MB): 7.6 2.7 4.9 Datab (blocc: 17.042 13.575 3.467,3 Tramit Buffer Cache (blocc: 5.898, 5.360, 537,6 Diretto (blocc: 11.143 8.214, 2.929,7 Questa tabella fornisce i seguenti valori per il throughput e gli IOPS totali: Throughput = 143 MBPS (dalla quinta riga, denominata Totale, seconda colonna)	

Operazione	Description	Competenze richieste
	IOPS = 4.367,4 (dalla prima riga, denominata Total Requests, seconda colonna)	
Opzione 4: utilizza le viste AWR.	È possibile visualizzare le stesse informazioni su IOPS e sulla velocità effettiva utilizzando le viste AWR. Per ottenere queste informazioni, utilizza la seguente query: `break on report compute sum of Value on report select METRIC_NAME,avg(AVERAGE) as "Value" from dba_hist_sysmetric_summary where METRIC_NAME in ('Physical Read Total IO Requests Per Sec','Physical Write Total IO Requests Per Sec') group by metric_name;`	DBA

Stima dei requisiti della CPU

Operazione	Description	Competenze richieste
Scegliere un metodo.	È possibile stimare la CPU richiesta per il database di destinazione in tre modi:	DBA

Operazione	Description	Competenze richieste
	• Utilizzando i core effettiva mente disponibili del processore • Utilizzando i core utilizzati in base alle statistiche del sistema operativo • Utilizzando i core utilizzati in base alle statistiche del database Se stai esaminando i core utilizzati, ti consigliamo di utilizzare il metodo delle metriche del database anziché le statistiche del sistema operativo, poiché si basa sulla CPU utilizzata solo dai database di cui intendi migrare. (Le statistiche del sistema operativo includono anche l'utilizzo della CPU da parte di altri processi). È inoltre necessario controllare i consigli relativi alla CPU nel rapporto ADDM per migliorar e le prestazioni dopo la migrazione. È inoltre possibile stimare i requisiti in base alla generazio ne della CPU. Se utilizzi diverse generazioni di CPU, puoi stimare la CPU richiesta per il database di destinazi	

Operazione	Description	Competenze richieste
	one seguendo le istruzion i contenute nel white paper Demystifying the Number of v for Optimal Workload Performance. CPUs	

Operazione	Description	Competenze richieste
Opzione 1: stima i requisiti in base ai core disponibili.	Nei report AWR: • CPUs si riferiscono a logico e virtuale CPUs. • I core sono il numero di processori in un chipset CPU fisico. • Un socket è un dispositivo fisico che collega un chip a una scheda. I processor i multi-core dispongono di socket con diversi core CPU. È possibile stimare i core disponibili in due modi: • Utilizzando i comandi del sistema operativo • Utilizzando il rapporto AWR Per stimare i core disponibi li utilizzando i comandi del sistema operativo Utilizzate il comando seguente per contare i core del processore. `$ cat /proc/cpuinfo | grep "cpu cores"|uniq cpu cores : 4 cat /proc/cpuinfo | egrep "core id|physic al id" | tr -d "\n" |`	DBA

Operazione	Description	Competenze richieste

	`sed s/physical/\\nphysical/g | grep -v ^$ | sort | uniq | wc -l`
	
	Utilizzate il seguente comando per contare i socket nel processore.
	
	`grep "physical id" /proc/cpuinfo | sort -u`
	`physical id : 0`
	`physical id : 1`
	
	Note
	Non è consigliabile utilizzare comandi del sistema operativo come nmon e sar per estrarre l'utilizzo della CPU. Questo perché tali calcoli includono l'utilizzo della CPU da parte di altri processi e potrebbero non riflettere l'effettiva CPU utilizzata dal database.
	
	Per stimare i core disponibili utilizzando il rapporto AWR
	
	È inoltre possibile ricavare l'utilizzo della CPU dalla prima sezione del rapporto AWR. Ecco un estratto del rapporto.	

Operazione	Description	Competenze richieste
	N ID Ista Inse Ora Vers RA D DB un di num avvi X <DB XX> 1 05 12,1 NO sett - 20 23:0 Hos Plat CPU Nuc Pres Mem Nar (Pia (GB) (No rma hos <ho Linu 80 80 2 441,7 e> x86 a 64 bit In questo esempio, il CPUs conteggio è 80, il che indica che sono logici (virtuali). CPUs È inoltre possibile notare che questa configurazione ha due socket, un processore fisico per socket (per un totale di due processori fisici) e 40 core per ogni processore o socket fisico.	

<table>
<tr><th>Operazione</th><th>Description</th><th>Competenze richieste</th></tr>
<tr><td>Opzione 2: stima dell'utilizzo della CPU utilizzando le statistiche del sistema operativo.</td><td>È possibile controllare le statistiche sull'utilizzo della CPU del sistema operativo direttamente nel sistema operativo (utilizzando sar o un'altra utilità del sistema operativo host) o esaminando i valori IDLE/ (IDLE+BUSY) dalla sezione Operating System Statistics del rapporto AWR. Puoi vedere i secondi di CPU consumati direttamente da v$osstat. I report AWR e Statspack mostrano questi dati anche nella sezione Statistiche del sistema operativo.

Se nella stessa casella sono presenti più database, tutti hanno gli stessi valori v$osstat per BUSY_TIME.

<table><tr><th>Statistic</th><th>Valore</th><th>Valore finale</th></tr><tr><td>FREE_MEMORY_BYT</td><td>6.810.67.248</td><td>12.280,799,232</td></tr><tr><td>INACTIV MEMOR TES</td><td>175.627.33.632</td><td>160,380,653,568</td></tr><tr><td>SWAP_F _BYTES</td><td>17.145.64.336</td><td>17,145,872,384</td></tr></table></td><td>DBA</td></tr>
</table>

Operazione	Description		Competenze richieste
	ORARIO DI LAVORO	1.305.56 .937	
	TEMPO DI INATTIVÀ	4.312.71 .839	
	IOWAIT-TIME	53.417.1 4	
	BEL MOMENT	29.815	
	SYS_TIM	148.567. 70	
	TEMPO_NTE	1.146.91 .783	
	CARICA	25	29
	VM_IN_BYTES	593.920	
	VM_OUTBYTES	327.680	
	BYTE_D MEMOR FISICA	474.362. 17.152	
	NUM_CP	80	
	NUM_CP ORES	80	

Operazione	Description		Competenze richieste
	NUM_CF OCKETS	2	
	GLOBAL RECEIVI IZE_MA)	4.194.30	
	DIMENS E MASSIM DI INVIO GLOBAL	2.097.15	
	TCP_RE VE_SIZE EFAULT	87.380	
	TCP_RE VE_SIZE AX	6.291.45	
	TCP_RE VE_SIZE IN	4,096	
	TCP_SE SIZE_DE ULT	16,384	
	TCP_SE SIZE_MA	4.194.30	
	TCP_SE SIZE_MI	4,096	

Operazione	Description	Competenze richieste
	Se nel sistema non sono presenti altri utenti principal i di CPU, utilizza la formula seguente per calcolare la percentuale di utilizzo della CPU: Utilizzo = Tempo di occupazio ne/Tempo totale Orario di lavoro = requisiti = V $OSStat.busy_time C = Tempo totale (occupato +inattivo) C = capacità = V$OSTAT.b usy_time + V$OSTAT.i dle_time Utilizzo = BUSY_TIME/ (BUSY_TIME + IDLE_TIME) = -1.305.569.937/(1.305.569.9 37 + 4.312.718.839) = 23% utilizzato	

Operazione	Description	Competenze richieste
Opzione 3: stima dell'utilizzo della CPU utilizzando le metriche del database.	Se nel sistema sono in esecuzione più database, puoi utilizzare le metriche del database visualizzate all'inizio del rapporto. Snap Id / Snap Time / Sess / Cursors session Inizia Snap: 1846 / 28 sette - 20 09:0(/ 1226 / 35,8 Fine Snap: 1854 / 06-2(ottob 13:0(/ 1876 / 41,1 Tras(: / 11,7! (minu Ora DB: / 312.(0 (minu Per ottenere i parametri di utilizzo della CPU, usa questa formula: Utilizzo della CPU del database (% della potenza della CPU disponibile) =	DBA

Operazione	Description	Competenze richieste
	tempo CPU/NUM_CPUS / tempo trascorso dove l'utilizzo della CPU è descritto in base al tempo impiegato dalla CPU e rappresenta il tempo impiegato sulla CPU, non il tempo di attesa della CPU. Questo calcolo si traduce in: = 312.625,40/11.759,64/80 = viene utilizzato il 33% della CPU Numero di core (33%) * 80 = 26,4 core Core totali = 26,4 * (120%) = 31,68 core Puoi utilizzare il maggiore di questi due valori per calcolare l'utilizzo della CPU dell'istanza Amazon RDS o Aurora DB. **Note** Su IBM AIX, l'utilizzo calcolato non corrisponde ai valori del sistema operativo o del database. Questi valori corrispondono su altri sistemi operativi.	

Stima dei requisiti di memoria

Operazione	Description	Competenze richieste
Stima i requisiti di memoria utilizzando le statistiche sulla memoria.	È possibile utilizzare il report AWR per calcolare la memoria del database di origine e confrontarla nel database di destinazione. È inoltre necessario verificare le prestazioni del database esistente e ridurre i requisiti di memoria per risparmia re sui costi o aumentare i requisiti per migliorare le prestazioni. Ciò richiede un'analisi dettagliata del tempo di risposta AWR e del contratto sul livello di servizio (SLA) dell'applicazione. Utilizza la somma dell'utilizzo dell'area globale del sistema (SGA) e dell'area globale del programma (PGA) di Oracle come utilizzo stimato della memoria per Oracle. Aggiungi un ulteriore 20 percento per il sistema operativo per determinare un requisito di dimensione della memoria di destinazione. Per Oracle RAC, utilizza la somma dell'utilizzo stimato della memoria su tutti i nodi RAC e riduci la memoria totale, poiché è archiviata su blocchi comuni.	DBA

Operazione	Description	Competenze richieste
	1. Controlla le metriche nella tabella Percentuale di efficienza delle istanze. La tabella usa i termini seguenti: • La percentuale di accessi al buffer è la percentua le di volte in cui un determinato blocco è stato trovato nella buffer cache anziché eseguire un I/O fisico. Per prestazioni migliori, scegli il 100%. • Buffer Nowait% dovrebbe essere vicino al 100 percento. • Latch Hit% dovrebbe essere vicino al 100 percento. • % Non-Parse CPU è la percentuale del tempo di CPU impiegato in attività non di analisi. Questo valore dovrebbe essere vicino al 100 percento. Percentuali di efficienz a delle istanze (obiettivo 100%) Buffer Nowa %: 99,99 Ripeti %: NoWa 100,00	

Operazione	Description	Competenze richieste
	% di succe al buffer 99,84 Ordin to% in memc 100,00 % di visite alla libreri 748,7 Analis morbi %: 99,81 Esegu per analiz e%: 96,61 Perce le di blocc 100,00 Analiz la CPU per Parse Elaps %: 72,73 % CPU non analiz a: 99,21 % di acces alla cache Flash 0,00 In questo esempio, tutte le metriche sembrano corrette, quindi puoi utilizzare SGA	

Operazione	Description	Competenze richieste
	e PGA per il database esistente come requisito di pianificazione della capacità. 2. Controllate la sezione delle statistiche sulla memoria e calcolate il valore SGA/PGA. Inizia / End Memoria host (MB): 452.387 / 452.387,3 Uso SGA (MB): 220.544 / 220544,0 Uso PGA (MB): 36.874,9 / 45.270,0 Memoria totale dell'istanza in uso = SGA + PGA = 220 GB+ 45 GB = 265 GB Aggiungi il 20 percento del buffer: Memoria totale dell'istanza = 1,2 * 265 GB = 318 GB Poiché SGA e PGA rappresentano il 70 percento della	

Operazione	Description	Competenze richieste
	memoria host, il fabbisogno totale di memoria è: Memoria host totale = 318/0,7 = 464 GB **Note** Quando esegui la migrazione ad Amazon RDS for Oracle, PGA e SGA vengono precalcolati in base a una formula predefinita. Assicurati che i valori precalcolati siano vicini alle tue stime.	

Determina il tipo di istanza DB del database di destinazione

Operazione	Description	Competenze richieste
Determina il tipo di istanza DB in base alle stime di I/O del disco, CPU e memoria.	In base alle stime dei passaggi precedenti, la capacità del database Amazon RDS o Aurora di destinazione dovrebbe essere: • 68 core di CPU • 143 MBPS di velocità effettiva • 4367 IOPS per I/O su disco • 464 GB di memoria	DBA

Operazione	Description	Competenze richieste
	Nel database Amazon RDS o Aurora di destinazione, puoi mappare questi valori al tipo di istanza db.r5.16xlarge, che ha una capacità di 32 core, 512 GB di RAM e 13.600 Mbps di velocità effettiva. Per ulteriori informazioni, consulta il post sul blog di AWS: istanze Amazon RDS di dimensioni corrette su larga scala in base ai parametri delle prestazioni Oracle.	

Risorse correlate

- Classe di istanza Aurora DB (documentazione Amazon Aurora)
- Storage di istanze database Amazon RDS (documentazione Amazon RDS)
- Strumento AWS Miner (GitHub repository)

Esporta tabelle Amazon RDS for SQL Server in un bucket S3 utilizzando AWS DMS

Subhani Shaik, Amazon Web Services

Riepilogo

Amazon Relational Database Service (Amazon RDS) per SQL Server non supporta il caricamento di dati su altri server collegati al motore DB sul cloud Amazon Web Services (AWS). Puoi invece utilizzare AWS Database Migration Service (AWS DMS) per esportare le tabelle Amazon RDS for SQL Server in un bucket Amazon Simple Storage Service (Amazon S3), dove i dati sono disponibili per altri motori DB.

AWS DMS ti aiuta a migrare i database in AWS in modo rapido e sicuro. Il database di origine rimane pienamente operativo durante la migrazione, riducendo al minimo i tempi di inattività delle applicazioni che si basano sul database. AWS DMS può migrare i dati da e verso i database commerciali e open source più utilizzati.

Questo modello utilizza AWS Secrets Manager durante la configurazione degli endpoint AWS DMS. Secrets Manager ti aiuta a proteggere i segreti necessari per accedere alle tue applicazioni, servizi e risorse IT. È possibile utilizzare il servizio per ruotare, gestire e recuperare le credenziali del database, le chiavi API e altri segreti durante il loro ciclo di vita. Gli utenti e le applicazioni recuperano i segreti con una chiamata a Secrets Manager, riducendo la necessità di codificare le informazioni sensibili. Secrets Manager offre una rotazione segreta con integrazione integrata per Amazon RDS, Amazon Redshift e Amazon DocumentDB. Inoltre, il servizio è estensibile ad altri tipi di segreti, tra cui chiavi e token API. OAuth Con Secrets Manager, puoi controllare l'accesso ai segreti utilizzando autorizzazioni granulari e controllare centralmente la rotazione segreta per le risorse nel cloud AWS, nei servizi di terze parti e in locale.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un bucket S3
- Un cloud privato virtuale (VPC)
- Una sottorete DB
- Amazon RDS per SQL Server

- Un ruolo AWS Identity and Access Management (IAM) con accesso (list, get e put objects) al bucket S3 per conto dell'istanza Amazon RDS.
- Secrets Manager per memorizzare le credenziali dell'istanza RDS.

Architecture

Stack tecnologico

- Amazon RDS per SQL Server
- AWS DMS
- Simple Storage Service (Amazon S3)
- AWS Secrets Manager

Architettura Target

Il diagramma seguente mostra l'architettura per l'importazione di dati dall'istanza Amazon RDS al bucket S3 con l'aiuto di AWS DMS.

1. L'attività di migrazione di AWS DMS: connessione all'istanza Amazon RDS di origine tramite l'endpoint di origine
2. Copia dei dati dall'istanza Amazon RDS di origine
3. L'attività di migrazione di AWS DMS che si connette al bucket S3 di destinazione tramite l'endpoint di destinazione
4. Esportazione dei dati copiati nel bucket S3 in formato CSV (valori separati da virgole)

Tools (Strumenti)

Servizi AWS

- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS.

- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Secrets Manager ti aiuta a sostituire le credenziali codificate nel codice, comprese le password, con una chiamata API a Secrets Manager per recuperare il segreto a livello di codice.

Altri servizi

- Microsoft SQL Server Management Studio (SSMS) è uno strumento per la gestione di SQL Server, che include l'accesso, la configurazione e l'amministrazione dei componenti di SQL Server.

Epiche

Configurazione dell'istanza Amazon RDS for SQL Server

Operazione	Description	Competenze richieste
Crea l'istanza Amazon RDS for SQL Server.	1. Apri la Console di gestione AWS, scegli RDS e utilizza l'opzione di creazione Standard per creare un'istanza Amazon RDS con l'edizione richiesta, come SQL Server Express Edition, SQL Server Standard Edition o SQL Server Enterprise Edition. Per la versione, scegli 2016 o successiva. 2. In Modelli, scegli Dev/Test.	DBA, ingegnere DevOps
Imposta le credenziali per l'istanza.	1. Immettete un nome per l'istanza. 2. Fornisci un nome utente e una password per l'istanza Amazon RDS.	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
Configura la classe, lo storage, la scalabilità automatica e la disponibilità dell'istanza.	1. Seleziona la classe di istanza DB dall'elenco: classi Standard, Memory Optimized e Burstable. Scegli il tipo di istanza DB che alloca la capacità di calcolo, di rete e di memoria richiesta dai carichi di lavoro pianificati per questa istanza DB. Per ulteriori informazioni, consulta la documentazione di AWS. 2. Seleziona il tipo di storage dall'elenco: General Purpose SSD, Provision ed IOPS SSD o Magnetic. Alloca la dimensione di archiviazione predefinita in base alle esigenze. 3. Scegli Enable storage autoscaling per aumentare lo storage Amazon RDS in base alla pianificazione della capacità. 4. Una distribuzione Multi-AZ con un'istanza di replica è supportata da AWS DMS. In caso di interruzione nella zona di disponibilità, nell'hardware interno o nella rete, AWS DMS creerà un'istanza di standby e fornirà l'alta disponibilità (HA) tramite	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
	il failover automatico sulle repliche di standby. A seconda delle dimensioni dell'importazione, seleziona l'opzione appropriata.	
Specificare il VPC, il gruppo di sottoreti, l'accesso pubblico e il gruppo di sicurezza.	Seleziona il VPC, i gruppi di sottorete DB e il gruppo di sicurezza VPC come richiesto per creare l'istanza Amazon RDS. Segui le best practice, ad esempio: • Non abilitate l'accesso pubblico all'istanza DB RDS. • Non utilizzare il CIDR 0.0.0.0/0 nei gruppi di sicurezza. • Utilizza solo l'indirizzo IP e i dettagli della porta richiesti per accedere all'istanza RDS.	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
Configura il monitoraggio, il backup e la manutenzione.	1. Specificate le opzioni di backup desiderate. Per impostazione predefinita, i backup automatici sono abilitati con un periodo di conservazione di 7 giorni. 2. Scegli le impostazioni appropriate per l'aggiornamento automatico della versione secondaria e la finestra di manutenzione per applicare le modifiche o la manutenzione in sospeso al database da parte di Amazon RDS. 3. Scegliere Crea database.	DBA, ingegnere DevOps

Configura il database e i dati di esempio

Operazione	Description	Competenze richieste
Crea una tabella e carica i dati di esempio.	Nel nuovo database, crea una tabella. Utilizzate il codice di esempio nella sezione Informazioni aggiuntive per caricare i dati nella tabella.	DBA, ingegnere DevOps

Imposta le credenziali

Operazione	Description	Competenze richieste
Crea il segreto.	1. Sulla console, scegli Secrets Manager e scegli Archivia un nuovo segreto. 2. Inserisci un nome utente e una password per il database Amazon RDS for SQL Server. Questo segreto verrà utilizzato per l'endpoint di origine AWS DMS.	DBA, ingegnere DevOps

Configura l'accesso tra il database e il bucket S3

Operazione	Description	Competenze richieste
Crea un ruolo IAM per l'accesso ad Amazon RDS.	1. Sulla console, scegli IAM e crea un ruolo IAM che consenta a un bucket S3 di read/write accedere ad Amazon RDS. 2. In Funzionalità, seleziona S3 Integration.	DBA, ingegnere DevOps

Crea il bucket S3

Operazione	Description	Competenze richieste
Crea il bucket S3.	Per salvare i dati da Amazon RDS for SQL Server, sulla console, scegli S3, quindi	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
	scegli Crea bucket. Assicurati che il bucket S3 non sia disponibile pubblicamente.	

Configura l'accesso tra AWS DMS e il bucket S3

Operazione	Description	Competenze richieste
Crea un ruolo IAM per AWS DMS per accedere ad Amazon S3.	Crea un ruolo IAM che consenta ad AWS DMS di elencare, ottenere e inserire oggetti dal bucket S3.	DBA, ingegnere DevOps

Configurazione di AWS DMS

Operazione	Description	Competenze richieste
Crea l'endpoint di origine AWS DMS.	1. Sulla console, scegli Database Migration Service e scegli Endpoints. Crea l'endpoint di origine, selezionando la casella di controllo Seleziona istanza DB RDS. 2. Per il motore Source, selezionare Microsoft SQL Server. 3. In Accesso al database degli endpoint, scegli AWS Secrets Manager e inserisci il ruolo segreto e IAM che hai creato in precedenza e il nome del database.	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
	4. Testa l'endpoint di origine.	
Crea l'endpoint di destinazione AWS DMS.	Crea l'endpoint Target, selezionando Amazon S3 come motore di Target. Fornisci il nome del bucket S3 e il nome della cartella per il ruolo IAM che hai creato in precedenza.	DBA, ingegnere DevOps
Crea l'istanza di replica AWS DMS.	Nello stesso VPC, sottorete e gruppo di sicurezza, crea l'istanza di replica AWS DMS. Per ulteriori informazioni sulla scelta di una classe di istanza, consulta la documentazione AWS.	DBA, ingegnere DevOps
Crea l'attività di migrazione AWS DMS.	Per esportare i dati da Amazon RDS for SQL Server al bucket S3, crea un'attività di migrazione del database. Per il tipo di migrazione, scegli Migra dati esistenti. Seleziona gli endpoint e l'istanza di replica AWS DMS che hai creato.	DBA, ingegnere DevOps

Esporta i dati nel bucket S3

Operazione	Description	Competenze richieste
Esegui l'attività di migrazione del database.	Per esportare i dati della tabella di SQL Server, avvia	DBA, ingegnere DevOps

Operazione	Description	Competenze richieste
	l'attività di migrazione del database. L'attività esporterà i dati da Amazon RDS for SQL Server nel bucket S3 in formato CSV.	

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Eliminare le risorse.	Per evitare costi aggiuntivi, utilizza la console per eliminare le risorse nell'ordine seguente: 1. Attività di migrazione 2. Istanza di replica 3. Endpoints 4. Bucket S3 5. Istanza di database	DBA, ingegnere DevOps

Risorse correlate

- AWS DMS
- Amazon S3
- Amazon RDS per SQL Server
- Integrazione con Amazon S3

Informazioni aggiuntive

Per creare il database e la tabella e caricare i dati di esempio, usa il codice seguente.

```
--Step1: Database creation in RDS SQL Server
```

```
CREATE DATABASE [Test_DB]
 ON  PRIMARY
( NAME = N'Test_DB', FILENAME = N'D:\rdsdbdata\DATA\Test_DB.mdf' , SIZE = 5120KB ,
 FILEGROWTH = 10%)
 LOG ON
( NAME = N'Test_DB_log', FILENAME = N'D:\rdsdbdata\DATA\Test_DB_log.ldf' , SIZE =
 1024KB , FILEGROWTH = 10%)
GO

--Step2: Create Table
USE Test_DB
GO
Create Table Test_Table(ID int, Company Varchar(30), Location Varchar(20))

--Step3: Load sample data.
USE Test_DB
GO
Insert into Test_Table values(1,'AnyCompany','India')
Insert into Test_Table values(2,'AnyCompany','USA')
Insert into Test_Table values(3,'AnyCompany','UK')
Insert into Test_Table values(4,'AnyCompany','Hyderabad')
Insert into Test_Table values(5,'AnyCompany','Banglore')
```

Gestisci blocchi anonimi nelle istruzioni SQL dinamiche in Aurora PostgreSQL

Nome utente, Amazon Web Services

Riepilogo

Nota: Amazon Cloud Directory non è più aperta a nuovi clienti. Per alternative a Cloud Directory, esplora Amazon DynamoDB e Amazon Neptune. Se hai bisogno di aiuto per scegliere l'alternativa giusta per il tuo caso d'uso o per qualsiasi altra domanda, contatta. Supporto AWS

Questo modello mostra come evitare l'errore che si verifica quando si gestiscono blocchi anonimi nelle istruzioni SQL dinamiche. Ricevi un messaggio di errore quando utilizzi AWS Schema Conversion Tool per convertire un database Oracle in un database Edition compatibile con Aurora PostgreSQL. Per evitare l'errore, devi conoscere il valore di una variabile 0UT bind, ma puoi conoscere il valore di una variabile 0UT bind solo dopo aver eseguito l'istruzione SQL. L'errore deriva dal fatto che AWS Schema Conversion Tool (AWS SCT) non comprende la logica all'interno dell'istruzione Dynamic SQL. AWS SCT non è in grado di convertire l'istruzione SQL dinamica in PL/SQL codice (ovvero funzioni, procedure e pacchetti).

Prerequisiti e limitazioni

Prerequisiti

- Account AWS attivo
- Istanza del database PostgreSQL (DB) Aurora
- Amazon Relational Database Service (Amazon RDS) per istanze database Oracle
- Terminale interattivo PostgreSQL (psql)
- SQL *Plus
- `AWS_ORACLE_EXT`schema (parte del pacchetto di estensione AWS SCT) nel database di destinazione
- Versione più recente di AWS Schema Conversion Tool (AWS SCT) e dei relativi driver richiesti

Architecture

Stack tecnologico di origine

- Oracle Database 10g locale e versione successiva

Stack tecnologico Target

- Amazon Aurora PostgreSQL
- Amazon RDS per PostgreSQL
- Strumento di conversione dello schema AWS (AWS SCT)

Architettura di migrazione

Il diagramma seguente mostra come utilizzare le variabili di 0UT associazione AWS SCT e Oracle per scansionare il codice dell'applicazione alla ricerca di istruzioni SQL incorporate e convertire il codice in un formato compatibile utilizzabile da un database Aurora.

Il diagramma mostra il flusso di lavoro seguente:

1. Genera un report AWS SCT per il database di origine utilizzando Aurora PostgreSQL come database di destinazione.
2. Identifica il blocco anonimo nel blocco di codice SQL dinamico (per il quale AWS SCT ha generato l'errore).
3. Converti il blocco di codice manualmente e distribuisci il codice su un database di destinazione.

Tools (Strumenti)

Servizi AWS

- Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL.
- Amazon Relational Database Service (Amazon RDS) per Oracle ti aiuta a configurare, gestire e scalare un database relazionale Oracle nel cloud AWS.
- AWS Schema Conversion Tool (AWS SCT) ti aiuta a rendere prevedibili le migrazioni di database eterogenei convertendo automaticamente lo schema del database di origine e la maggior parte degli oggetti di codice del database in un formato compatibile con il database di destinazione.

Altri strumenti

- pgAdmin consente di connettersi e interagire con il server di database.

- Oracle SQL Developer è un ambiente di sviluppo integrato che è possibile utilizzare per sviluppare e gestire database in Oracle Database. È possibile utilizzare SQL *Plus o Oracle SQL Developer per questo modello.

Epiche

Configura il database di origine Oracle

Operazione	Description	Competenze richieste
Crea un'istanza Oracle su Amazon RDS o Amazon EC2.	Per creare un'istanza DB Oracle su Amazon RDS, consulta Creazione di un'istanza DB Oracle e connessione a un database su un'istanza Oracle DB nella documentazione di Amazon RDS. Per creare un'istanza DB Oracle su Amazon Elastic Compute Cloud (Amazon EC2), consulta Amazon EC2 for Oracle nella documenta zione di AWS Prescriptive Guidance.	DBA
Crea uno schema di database e oggetti per la migrazione.	Puoi usare Amazon Cloud Directory per creare uno schema di database. Per ulteriori informazioni, consulta Create a Schema nella documentazione di Cloud Directory.	DBA
Configura i gruppi di sicurezza in entrata e in uscita.	Per creare e configurare gruppi di sicurezza, consulta Controllare l'accesso con	DBA

Operazione	Description	Competenze richieste
	i gruppi di sicurezza nella documentazione di Amazon RDS.	
Conferma che il database sia in esecuzione.	Per verificare lo stato del database, consulta Visualizzazione degli eventi di Amazon RDS nella documentazione di Amazon RDS.	DBA

Configurare il database Aurora PostgreSQL di destinazione

Operazione	Description	Competenze richieste
Crea un'istanza Aurora PostgreSQL in Amazon RDS.	Per creare un'istanza Aurora PostgreSQL, consulta Creazione di un cluster DB e connessione a un database su un cluster Aurora PostgreSQL DB nella documentazione di Amazon RDS.	DBA
Configura un gruppo di sicurezza in entrata e in uscita.	Per creare e configurare gruppi di sicurezza, consulta Fornire l'accesso al cluster DB nel VPC creando un gruppo di sicurezza nella documentazione di Aurora.	DBA
Verificare che il database Aurora PostgreSQL sia in esecuzione.	Per verificare lo stato del database, consulta Visualizzazione degli eventi di Amazon RDS nella documentazione di Aurora.	DBA

Configurazione di AWS SCT

Operazione	Description	Competenze richieste
Connect AWS SCT al database di origine.	Per connettere AWS SCT al tuo database di origine, consulta Connessione a PostgreSQL come sorgente nella documentazione di AWS SCT.	DBA
Connect AWS SCT al database di destinazione.	Per connettere AWS SCT al tuo database di destinazione, consulta What is the AWS Schema Conversion Tool? nella Guida per l'utente di AWS Schema Conversion Tool.	DBA
Converti lo schema del database in AWS SCT e salva il codice convertito automatic amente come file SQL.	Per salvare i file convertiti da AWS SCT, consulta Salvare e applicare lo schema convertit o in AWS SCT nella Guida per l'utente di AWS Schema Conversion Tool.	DBA

Migrare il codice

Operazione	Description	Competenze richieste
Scarica il file SQL per la conversione manuale.	Nel file convertito da AWS SCT, estrai il file SQL che richiede la conversione manuale.	DBA
Aggiorna lo script.	Aggiorna manualmente il file SQL.	DBA

Risorse correlate

- Amazon RDS
- Caratteristiche di Amazon Aurora

Informazioni aggiuntive

Il codice di esempio seguente mostra come configurare il database di origine Oracle:

```
CREATE or replace PROCEDURE calc_stats_new1 (
  a NUMBER,
  b NUMBER,
  result out NUMBER)
IS
BEGIN
result:=a+b;
END;
/
```

```
set serveroutput on ;

DECLARE
  a NUMBER := 4;
  b NUMBER := 7;
  plsql_block VARCHAR2(100);
  output number;
BEGIN
  plsql_block := 'BEGIN calc_stats_new1(:a, :b,:output); END;';
  EXECUTE IMMEDIATE plsql_block USING a, b,out output;
  DBMS_OUTPUT.PUT_LINE('output:'||output);

END;
```

Il codice di esempio seguente mostra come configurare il database di destinazione Aurora PostgreSQL:

```
 w integer,
 x integer)
RETURNS integer
AS
$BODY$
```

```
DECLARE
begin
return w + x ;
end;
$BODY$
LANGUAGE  plpgsql;


CREATE OR REPLACE FUNCTION test_pg.init()
RETURNS void
AS
$BODY$
BEGIN
if aws_oracle_ext.is_package_initialized
      ('test_pg' ) then
      return;
    end if;
    perform aws_oracle_ext.set_package_initialized
      ('test_pg' );

PERFORM aws_oracle_ext.set_package_variable('test_pg', 'v_output', NULL::INTEGER);
PERFORM aws_oracle_ext.set_package_variable('test_pg', 'v_status', NULL::text);
END;
$BODY$
LANGUAGE  plpgsql;


DO $$
declare
v_sql text;
v_output_loc int;
a integer :=1;
b integer :=2;
BEGIN
perform  test_pg.init();
--raise notice 'v_sql %',v_sql;
execute 'do $a$ declare v_output_l int; begin select * from test_pg.calc_stats_new1('||
a||','||b||') into v_output_l;
PERFORM aws_oracle_ext.set_package_variable(''test_pg'', ''v_output'', v_output_l) ;
 end; $a$'  ;
v_output_loc := aws_oracle_ext.get_package_variable('test_pg', 'v_output');
raise notice 'v_output_loc %',v_output_loc;
END ;
```

```
$$
```

Aiutaci a far rispettare il tagging di DynamoDB

Mansi Suratwala, Amazon Web Services

Riepilogo

Questo modello imposta notifiche automatiche quando un tag Amazon DynamoDB predefinito manca o viene rimosso da una risorsa DynamoDB sul cloud Amazon Web Services (AWS).

DynamoDB è un servizio di database NoSQL completamente gestito che offre prestazioni veloci e prevedibili con scalabilità. DynamoDB consente di alleggerire gli oneri amministrativi legati al funzionamento e alla scalabilità di un database distribuito. Quando utilizzi DynamoDB, non devi preoccuparti del provisioning, dell'installazione e della configurazione dell'hardware, della replica, dell'applicazione di patch software o della scalabilità del cluster.

Il modello utilizza un CloudFormation modello AWS, che crea un evento Amazon CloudWatch Events e una funzione AWS Lambda. L'evento rileva eventuali informazioni di tagging nuove o esistenti su DynamoDB utilizzando AWS. CloudTrail Se un tag predefinito è mancante o rimosso, CloudWatch attiva una funzione Lambda, che ti invia una notifica Amazon Simple Notification Service (Amazon SNS) che ti informa della violazione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un bucket Amazon Simple Storage Service (Amazon S3) per il file.zip Lambda che contiene lo script Python per l'esecuzione della funzione Lambda

Limitazioni

- La soluzione funziona solo quando si verificano gli eventi o. `TagResource` `UntagResource` CloudTrail Non crea notifiche per altri eventi.

Architecture

Stack tecnologico Target

- Amazon DynamoDB

- AWS CloudTrail
- Amazon CloudWatch
- AWS Lambda
- Simple Storage Service (Amazon S3)
- Amazon SNS

Architettura Target

Automazione e scalabilità

Puoi utilizzare il CloudFormation modello AWS più volte per diverse regioni e account AWS. È necessario eseguire il modello solo una volta in ogni regione o account.

Tools (Strumenti)

Strumenti

- Amazon DynamoDB — DynamoDB è un servizio di database NoSQL completamente gestito che offre prestazioni veloci e prevedibili con scalabilità.
- AWS CloudTrail: CloudTrail è un servizio AWS che ti aiuta con la governance, la conformità e il controllo operativo e dei rischi del tuo account AWS. Le azioni intraprese da un utente, un ruolo o un servizio AWS vengono registrate come eventi in CloudTrail.
- Amazon CloudWatch Events: Amazon CloudWatch Events offre un flusso quasi in tempo reale di eventi di sistema che descrivono i cambiamenti nelle risorse AWS.
- AWS Lambda — Lambda è un servizio di elaborazione che supporta l'esecuzione di codice senza la necessità di fornire o gestire server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti altamente scalabile che può essere utilizzato per un'ampia gamma di soluzioni di storage, tra cui siti Web, applicazioni mobili, backup e data lake.
- Amazon SNS — Amazon Simple Notification Service (Amazon SNS) è un servizio Web che consente alle applicazioni, agli utenti finali e ai dispositivi di inviare e ricevere istantaneamente notifiche dal cloud.

Codice

- Un file.zip del progetto è disponibile come allegato.

Epiche

Definisci il bucket S3

Operazione	Description	Competenze richieste
Definisci il bucket S3.	Sulla console Amazon S3, scegli o crea un bucket S3 con un nome univoco che non contenga barre iniziali. Questo bucket S3 ospiterà il file.zip con codice Lambda. Il bucket S3 deve trovarsi nella stessa regione AWS della risorsa DynamoDB monitorata.	Architetto del cloud

Carica il codice Lambda nel bucket S3

Operazione	Description	Competenze richieste
Carica il codice Lambda nel bucket S3.	Carica il file.zip con codice Lambda fornito nella sezione Allegati nel bucket S3. Il bucket S3 deve trovarsi nella stessa regione della risorsa DynamoDB monitorata.	Architetto del cloud

Implementa il modello AWS CloudFormation

Operazione	Description	Competenze richieste
Implementa il CloudFormation modello AWS.	Sulla CloudFormation console AWS, distribuisci il CloudForm	Architetto del cloud

Operazione	Description	Competenze richieste
	ation modello AWS fornito nella sezione Allegati. Nella prossima epopea, fornisci i valori per i parametri.	

Completa i parametri nel CloudFormation modello AWS

Operazione	Description	Competenze richieste
Assegna un nome al bucket S3.	Inserisci il nome del bucket S3 che hai creato o scelto nella prima epopea.	Architetto del cloud
Fornisci la chiave Amazon S3.	Fornisci la posizione del file.zip del codice Lambda nel tuo bucket S3, senza barre iniziali (ad esempio,) . `<folder>/<file-name>.zip`	Architetto del cloud
Fornisci un indirizzo email	Fornisci un indirizzo e-mail attivo per ricevere le notifiche di Amazon SNS.	Architetto del cloud
Definisci il livello di registrazione.	Definisci il livello e la frequenza di registrazione per la tua funzione Lambda. `Info`indica messaggi informativi dettagliati sullo stato di avanzamento dell'applicazione. `Error`indica eventi di errore che potrebbero comunque consentire all'applicazione di continuare a funzionare. `Warning`indica	Architetto del cloud

Operazione	Description	Competenze richieste
	situazioni potenzialmente dannose.	
Immettete le chiavi dei tag DynamoDB richieste.	Assicurati che i tag siano separati da virgole, senza spazi tra loro (ad esempio,) . `ApplicationId,CreatedBy,Environment,Organization` L'evento CloudWatch Events cerca questi tag e invia una notifica se non vengono trovati.	Architetto del cloud

Confermare la sottoscrizione.

Operazione	Description	Competenze richieste
Confermare la sottoscrizione.	Quando il modello viene distribuito correttamente, invia un'e-mail di iscrizione all'indirizzo e-mail che hai fornito. Per ricevere notifiche di violazione, devi confermare questa sottoscrizione e-mail.	Architetto del cloud

Risorse correlate

- Creazione di un bucket S3
- Caricamento di file in un bucket S3
- Taggare le risorse in DynamoDB
- Creazione di una regola CloudWatch Events che si attiva su una chiamata API AWS utilizzando AWS CloudTrail

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Implementa il disaster recovery tra regioni con AWS DMS e Amazon Aurora

Mark Hudson, Amazon Web Services

Riepilogo

I disastri naturali o causati dall'uomo possono verificarsi in qualsiasi momento e possono influire sulla disponibilità di servizi e carichi di lavoro in esecuzione in una determinata regione AWS. Per mitigare i rischi, devi sviluppare un piano di disaster recovery (DR) che incorpori le funzionalità interregionali integrate dei servizi AWS. Per i servizi AWS che non forniscono intrinsecamente funzionalità interregionali, il piano DR deve fornire anche una soluzione per gestire il failover tra le regioni AWS.

Questo modello ti guida attraverso una configurazione di disaster recovery che coinvolge due cluster di database Edition compatibili con Amazon Aurora MySQL in un'unica regione. Per soddisfare i requisiti di DR, i cluster di database sono configurati per utilizzare la funzionalità di database globale di Amazon Aurora, con un singolo database che si estende su più regioni AWS. Un task di AWS Database Migration Service (AWS DMS) replica i dati tra i cluster nella regione locale. AWS DMS, tuttavia, attualmente non supporta il failover delle attività tra regioni. Questo modello include i passaggi necessari per aggirare tale limitazione e configurare in modo indipendente AWS DMS in entrambe le regioni.

Prerequisiti e limitazioni

Prerequisiti

- Regioni AWS primarie e secondarie selezionate che supportano i database globali Amazon Aurora.
- Due cluster di database indipendenti della versione compatibile con Amazon Aurora MySQL in un unico account nella regione principale.
- Classe di istanza di database db.r5 o superiore (consigliata).
- Un'attività AWS DMS nella regione principale che esegue la replica continua tra i cluster di database esistenti.
- Risorse della regione DR disponibili per soddisfare i requisiti per la creazione di istanze di database. Per ulteriori informazioni, consulta Lavorare con un'istanza DB in un VPC.

Limitazioni

- Per l'elenco completo delle limitazioni dei database globali di Amazon Aurora, consulta Limitazioni dei database globali di Amazon Aurora.

Versioni del prodotto

- Amazon Aurora compatibile con MySQL Edition 5.7 o 8.0. Per ulteriori informazioni, consulta le versioni di Amazon Aurora.

Architecture

Stack tecnologico Target

- Cluster di database globale Amazon Aurora compatibile con MySQL Edition
- AWS DMS

Architettura Target

Il diagramma seguente mostra un database globale per due regioni AWS, una con i database principali e reporter principali e la replica AWS DMS, e una con i database secondari principali e reporter.

Automazione e scalabilità

Puoi usare AWS CloudFormation per creare l'infrastruttura prerequisita nella regione secondaria, come il cloud privato virtuale (VPC), le sottoreti e i gruppi di parametri. Puoi anche usare AWS CloudFormation per creare i cluster secondari nella regione DR e aggiungerli al database globale. Se hai utilizzato CloudFormation modelli per creare i cluster di database nella regione primaria, puoi aggiornarli o ampliarli con un modello aggiuntivo per creare la risorsa di database globale. Per ulteriori informazioni, consulta Creazione di un cluster Amazon Aurora DB con due istanze DB e Creazione di un cluster di database globale per Aurora MySQL.

Infine, puoi creare le attività AWS DMS nelle regioni primarie e secondarie utilizzando CloudFormation dopo che si sono verificati eventi di failover e failback. Per ulteriori informazioni, consulta AWS::DMS::ReplicationTask.

Tools (Strumenti)

- Amazon Aurora è un motore di database relazionale completamente gestito compatibile con MySQL e PostgreSQL. Questo modello utilizza l'edizione compatibile con Amazon Aurora MySQL.
- I database globali di Amazon Aurora sono progettati per applicazioni distribuite a livello globale. Un singolo database globale di Amazon Aurora può estendersi su più regioni AWS. Replica i dati senza alcun impatto sulle prestazioni del database. Consente inoltre letture locali veloci con bassa latenza in ogni regione e fornisce il disaster recovery in caso di interruzioni a livello regionale.
- AWS DMS offre una migrazione una tantum o una replica continua. Un'attività di replica continua mantiene sincronizzati i database di origine e di destinazione. Dopo la configurazione, l'attività di replica in corso applica continuamente le modifiche all'origine alla destinazione con una latenza minima. Tutte le funzionalità di AWS DMS, come la convalida e le trasformazioni dei dati, sono disponibili per qualsiasi attività di replica.

Epiche

Prepara i cluster di database esistenti nella regione principale

Operazione	Description	Competenze richieste
Modifica il gruppo di parametri del cluster di database.	Nel gruppo di parametri del cluster di database esistente, attiva la registrazione binaria a livello di riga impostando il **`binlog_format`** parametro su un valore di row. AWS DMS richiede la registrazione binaria a livello di riga per i database compatibili con MySQL durante l'esecuzione di repliche continue o l'acquisizione di dati di modifica (CDC). Per ulteriori informazioni, consulta Usare un database compatibile con	Amministratore AWS

Operazione	Description	Competenze richieste
	MySQL gestito da AWS come fonte per AWS DMS.	

Operazione	Description	Competenze richieste

Aggiorna il periodo di conservazione dei log binari del database.	Utilizzando un client MySQL installato sul dispositivo dell'utente finale o un'istanz a Amazon Elastic Compute Cloud (EC2Amazon), esegui la seguente procedura memorizzata fornita da Amazon Relational Database Service (Amazon RDS) sul nodo di scrittura del cluster di database principaleXX, dove è il numero di ore per conservar e i log.
	`call mysql.rds_set_conf iguration('binlog retention hours', XX)`
	Conferma l'impostazione eseguendo il comando seguente.
	`call mysql.rds_show_con figuration;`
	I database compatibili con MySQL gestiti da AWS eliminano i log binari il prima possibile. Pertanto, il periodo di conservazione deve essere sufficientemente lungo da garantire che i log non vengano eliminati prima dell'esecuzione del task AWS DMS. Un valore di 24 ore è
		DBA

Operazione	Description	Competenze richieste
	in genere sufficiente, ma il valore deve basarsi sul tempo necessario per configura re l'attività AWS DMS nella regione DR.	

Aggiorna l'attività AWS DMS esistente nella regione principale

Operazione	Description	Competenze richieste

Registra l'ARN dell'attività AWS DMS.	Utilizza Amazon Resource Name (ARN) per ottenere il nome dell'attività AWS DMS per un uso successivo. Per recuperare l'ARN del task AWS DMS, visualizza l'attivit à nella console o esegui il comando seguente.
	`aws dms describe-replication-tasks`
	Un ARN ha il seguente aspetto.
	`arn:aws:dms:us-eas t-1:<accountid>:ta sk:AN6HFFMPM246XOZ VEUHCNSOVF7MQCLTOZ UIRAMY`
	I caratteri dopo l'ultimo punto corrispondono al nome dell'atti vità utilizzato in un passaggio successivo.
		Amministratore AWS

Operazione	Description	Competenze richieste
Modifica l'attività AWS DMS esistente per registrare il checkpoint.	AWS DMS crea checkpoint che contengono informazioni in modo che il motore di replica conosca il punto di ripristino per il flusso di modifiche. Per registrare le informazioni sui checkpoint, esegui i seguenti passaggi nella console: 1. Interrompi l'attività AWS DMS. 2. Utilizza l'editor JSON nell'attività per impostare il TaskRecoveryTableEnabled parametro su true. 3. Avvia il task AWS DMS.	Amministratore AWS

Operazione	Description	Competenze richieste
Convalida le informazioni sui checkpoint.	Utilizzando un client MySQL connesso all'endpoint writer per il cluster, interroga la nuova tabella di metadati nel cluster di database reporter per verificare che esista e contenga le informazioni sullo stato di replica. Eseguire il seguente comando seguente. `select * from awsdms_control.awsdms_txn_state;` Il nome dell'attività dell'ARN deve essere trovato in questa tabella nella `Task_Name` colonna.	DBA

Espandi entrambi i cluster Amazon Aurora in una regione DR

Operazione	Description	Competenze richieste
Crea un'infrastruttura di base nella regione DR.	Crea i componenti di base necessari per la creazione e l'accesso ai cluster Amazon Aurora: • Virtual Private Cloud (VPC) (Cloud privato virtuale (VPC) • Sottoreti • Gruppo di sicurezza	Amministratore AWS

Operazione	Description	Competenze richieste
	• Liste di controllo degli accessi alla rete • Gruppo di sottoreti • DB parameter group (Gruppo di parametri database) • DB cluster parameter group (Gruppo di parametri del cluster database) Assicuratevi che la configura zione di entrambi i gruppi di parametri corrisponda alla configurazione nella regione principale.	
Aggiungi la regione DR a entrambi i cluster Amazon Aurora.	Aggiungi una regione secondaria (la regione DR) ai cluster Amazon Aurora principali e reporter. Per ulteriori informazioni, consulta Aggiungere una regione AWS a un database globale Amazon Aurora.	Amministratore AWS

Esegui il failover

Operazione	Description	Competenze richieste
Interrompi l'attività AWS DMS.	L'attività AWS DMS nella regione primaria non funzioner à correttamente dopo il	Amministratore AWS

Operazione	Description	Competenze richieste
	failover e deve essere interrott a per evitare errori.	
Esegui un failover gestito.	Eseguire un failover gestito del cluster di database principale nella regione DR. Per istruzion i, consulta Esecuzione di failover pianificati gestiti per i database globali di Amazon Aurora. Una volta completato il failover sul cluster di database principale, esegui la stessa attività sul cluster di database reporter.	Amministratore AWS, DBA
Carica i dati nel database principale.	Inserisci i dati di test nel nodo writer del database principal e nel cluster di database DR. Questi dati verranno utilizzat i per verificare che la replica funzioni correttamente.	DBA
Crea l'istanza di replica AWS DMS.	Per creare l'istanza di replica AWS DMS nella regione DR, consulta Creazione di un'istanza di replica.	Amministratore AWS, DBA

Operazione	Description	Competenze richieste
Crea gli endpoint di origine e destinazione di AWS DMS.	Per creare gli endpoint di origine e destinazione di AWS DMS nella regione DR, consulta Creazione di endpoint di origine e destinazione. L'origine deve puntare all'istanza writer del cluster di database principale. La destinazione deve puntare all'istanza writer del cluster di database reporter.	Amministratore AWS, DBA
Ottieni il checkpoint di replica.	Per ottenere il checkpoint di replica, utilizzate un client MySQL per interrogare la tabella dei metadati eseguendo quanto segue sul nodo writer nel cluster di database reporter nella regione DR. `select * from awsdms_control.awsdms_txn_state;` Nella tabella, trova il valore task_name che corrisponde all'ARN del task AWS DMS esistente nella regione primaria che hai ottenuto nella seconda epic.	DBA

Operazione	Description	Competenze richieste
Crea un task AWS DMS.	Utilizzando la console, crea un'attività AWS DMS nella regione DR. Nell'attività, specifica un metodo di migrazione di Replicate data changes only. Per ulteriori informazioni, vedere Creazione di un'attività. 1. Nelle impostazioni dell'attività, utilizza la procedura guidata per specificare quanto segue: • Modalità di avvio CDC per le transazioni di origine: abilita la modalità di avvio CDC personalizzata • Punto iniziale CDC personalizzato per le transazioni di origine: specifica un checkpoint di ripristino 2. Nella casella di controllo Recovery, inserisci il valore del checkpoint di replica precedentemente ottenuto tramite la query del database sulla tabella. `awsdms_txn_state` 3. Nella sezione delle impostazioni delle attività, seleziona l'editor JSON e imposta il TaskRecov	Amministratore AWS, DBA

Operazione	Description	Competenze richieste
	eryTableEnabledparametro su true. Imposta l'impostazione dell'attività di migrazione AWS DMS task Start su Automatically on create.	
Registra l'ARN dell'attività AWS DMS.	Usa l'ARN per ottenere il nome del task AWS DMS per un uso successivo. Per recuperare l'ARN del task AWS DMS, esegui il comando seguente. `aws dms describe-replication-tasks`	Amministratore AWS, DBA
Convalida i dati replicati.	Interroga il cluster di database reporter nella regione DR per confermare che i dati di test caricati nel cluster di database principale siano stati replicati.	DBA

Esegui il failback

Operazione	Description	Competenze richieste
Interrompi l'attività AWS DMS.	L'attività AWS DMS nella regione DR non funzionerà correttamente dopo il failback e deve essere interrotta per evitare errori.	Amministratore AWS

Operazione	Description	Competenze richieste
Esegui un failback gestito.	Esegui il failback del cluster di database principale nella regione principale. Per istruzioni, consulta Esecuzione di failover pianificati gestiti per i database globali di Amazon Aurora. Una volta completato il failback sul cluster di database principale, esegui la stessa attività sul cluster di database reporter.	Amministratore AWS, DBA
Ottieni il checkpoint di replica.	Per ottenere il checkpoin t di replica, utilizzate un client MySQL per interroga re la tabella dei metadati eseguendo quanto segue sul nodo writer nel cluster di database reporter nella regione DR. `select * from awsdms_co ntrol.awsdms_txn_s tate;` Nella tabella, trova il `task_name` valore che corrisponde all'ARN del task AWS DMS esistente nella regione DR che hai ottenuto nella quarta epopea.	DBA

Operazione	Description	Competenze richieste
Aggiorna gli endpoint di origine e destinazione di AWS DMS.	Dopo il failback dei cluster di database, controlla i cluster nella regione primaria per determinare quali nodi sono le istanze di scrittura. Verifica quindi che gli endpoint di origine e destinazione di AWS DMS esistenti nella regione primaria puntino alle istanze writer. In caso contrario, aggiorna gli endpoint con i nomi DNS (Domain Name System) dell'istanza di scrittura.	Amministratore AWS

Operazione	Description	Competenze richieste
Crea un task AWS DMS.	Utilizzando la console, crea un'attività AWS DMS nella regione principale. Nell'attività, specifica un metodo di migrazione di Replicate data changes only. Per ulteriori informazioni, vedere Creazione di un'attività. 1. Nelle impostazioni dell'attività, utilizza la procedura guidata e specifica quanto segue: • Modalità di avvio CDC per le transazioni di origine: abilita la modalità di avvio CDC personalizzata • Punto iniziale CDC personalizzato per le transazioni di origine: specifica un checkpoint di ripristino 2. Nella casella di controllo Recovery, inserisci il valore del checkpoint di replica precedentemente ottenuto tramite la query del database sulla tabella. `awsdms_txn_state` 3. Inoltre, nella sezione delle impostazioni delle attività, seleziona l'editor JSON e imposta il TaskRecov	Amministratore AWS, DBA

Operazione	Description	Competenze richieste
	eryTableEnabledparametro su true. 4. Infine, imposta l'imposta zione dell'attività di migrazione AWS DMS task Start su Automatically on create.	
Registra l'attività AWS DMS Amazon Resource Name (ARN).	Usa l'ARN per ottenere il nome del task AWS DMS per un uso successivo. Per recuperare l'ARN del task AWS DMS, esegui il seguente comando: `aws dms describe-replication-tasks` Il nome dell'attività sarà necessario quando si esegue un altro failover gestito o durante uno scenario di DR.	Amministratore AWS, DBA
Eliminare le attività di AWS DMS.	Elimina l'attività AWS DMS originale (attualmente interrotta) nella regione principale e l'attività AWS DMS esistente (attualmente interrotta) nella regione secondaria.	Amministratore AWS

Risorse correlate

- Configurazione del cluster Amazon Aurora DB
- Using Amazon Aurora global databases

- Lavorare con Amazon Aurora MySQL
- Utilizzo di un'istanza di replica AWS DMS
- Utilizzo degli endpoint AWS DMS
- Lavorare con le attività di AWS DMS
- Che cos'è AWS CloudFormation?

Informazioni aggiuntive

I database globali di Amazon Aurora vengono utilizzati in questo esempio per il DR perché forniscono un obiettivo del tempo di ripristino (RTO) efficace di 1 secondo e un obiettivo del punto di ripristino (RPO) inferiore a 1 minuto, entrambi inferiori rispetto alle soluzioni replicate tradizionali e ideali per scenari di DR.

I database globali di Amazon Aurora offrono molti altri vantaggi, tra cui:

- Letture globali con latenza locale: i consumatori globali possono accedere alle informazioni in una regione locale, con latenza locale.
- Cluster Amazon Aurora DB secondari scalabili: i cluster secondari possono essere scalati indipendentemente, aggiungendo fino a 16 repliche di sola lettura.
- Replica rapida dai cluster Amazon Aurora DB primari a quelli secondari: la replica ha un impatto minimo sulle prestazioni del cluster primario. Si verifica a livello di storage, con latenze di replica tipiche tra regioni inferiori a 1 secondo.

Questo modello utilizza anche AWS DMS per la replica. I database Amazon Aurora offrono la possibilità di creare repliche di lettura, che possono semplificare il processo di replica e la configurazione del DR. Tuttavia, AWS DMS viene spesso utilizzato per la replica quando sono necessarie trasformazioni dei dati o quando il database di destinazione richiede indici aggiuntivi che il database di origine non dispone.

Esegui la migrazione di funzioni e procedure Oracle con più di 100 argomenti a PostgreSQL

Srinivas Potlachervoo, Amazon Web Services

Riepilogo

Questo modello mostra come migrare le funzioni e le procedure di Oracle Database con più di 100 argomenti a PostgreSQL. Ad esempio, puoi utilizzare questo modello per migrare le funzioni e le procedure Oracle verso uno dei seguenti servizi di database AWS compatibili con PostgreSQL:

- Amazon Relational Database Service (Amazon RDS) per PostgreSQL
- Amazon Aurora PostgreSQL-Compatible Edition

PostgreSQL non supporta funzioni o procedure con più di 100 argomenti. Come soluzione alternativa, puoi definire un nuovo tipo di dati con campi di tipo che corrispondono agli argomenti della funzione di origine. È quindi possibile creare ed eseguire una PL/pgSQL funzione che utilizza il tipo di dati personalizzato come argomento.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un'istanza di database Amazon RDS Oracle (DB)
- Un'istanza DB Amazon RDS per PostgreSQL o un'istanza DB Aurora compatibile con PostgreSQL

Versioni del prodotto

- Istanza Amazon RDS Oracle DB versioni 10.2 e successive
- Istanza DB Amazon RDS PostgreSQL 9.4 e successive o istanze DB compatibili con Aurora PostgreSQL versioni 9.4 e successive
- Oracle SQL Developer versione 18 e successive
- pGAdmin versione 4 e successive

Architecture

Stack tecnologico di origine

- Istanza Amazon RDS Oracle DB versioni 10.2 e successive

Stack tecnologico Target

- Istanza DB Amazon RDS PostgreSQL 9.4 e successive o istanze DB compatibili con Aurora PostgreSQL versioni 9.4 e successive

Tools (Strumenti)

Servizi AWS

- Amazon Relational Database Service (Amazon RDS) per PostgreSQL ti aiuta a configurare, gestire e scalare un database relazionale PostgreSQL nel cloud AWS.
- Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL.

Altri servizi

- Oracle SQL Developer è un ambiente di sviluppo integrato che semplifica lo sviluppo e la gestione dei database Oracle nelle implementazioni tradizionali e basate sul cloud.
- pgAdmin è uno strumento di gestione open source per PostgreSQL. Fornisce un'interfaccia grafica che consente di creare, gestire e utilizzare oggetti di database.

Best practice

Assicuratevi che il tipo di dati che create corrisponda ai campi di tipo inclusi nella funzione o procedura Oracle di origine.

Epiche

Esegui una funzione o una procedura Oracle con più di 100 argomenti

Operazione	Description	Competenze richieste
Crea o identifica una Oracle/PLSQL funzione o una procedura esistente con più di 100 argomenti.	Create una Oracle/PLSQL funzione o una procedura con più di 100 argomenti. oppure Identifica una Oracle/PLSQL funzione o una procedura esistente con più di 100 argomenti. Per ulteriori informazioni, vedere le sezioni 14.7 CREATE FUNCTION Statement e 14.11 CREATE PROCEDURE Statement nella documentazione del database Oracle.	Conoscenza di Oracle/PLSQL
Compilare la funzione o la procedura. Oracle/PLSQL	Compilare la Oracle/PLSQL funzione o la procedura. Per ulteriori informazioni, vedere Compilazione di una funzione nella documenta zione di Oracle Database.	Conoscenza di Oracle/PLSQL
Esegui la funzione. Oracle/PLSQL	Esegui la Oracle/PLSQL funzione o la procedura. Quindi, salvate l'output.	Conoscenza di Oracle/PLSQL

Definire un nuovo tipo di dati che corrisponda agli argomenti della funzione o della procedura di origine

Operazione	Description	Competenze richieste
Definisci un nuovo tipo di dati in PostgreSQL.	Definisci un nuovo tipo di dati in PostgreSQL che includa tutti gli stessi campi che compaiono negli argomenti della funzione o della procedura Oracle di origine. Per ulteriori informazioni, vedere CREATE TYPE nella documentazione di PostgreSQL.	Conoscenza di PostgreSQL PL/pgSQL

Crea una funzione PostgreSQL che includa il nuovo argomento TYPE

Operazione	Description	Competenze richieste
Crea una funzione PostgreSQL che includa il nuovo tipo di dati.	Crea una funzione PostgreSQL che includa il nuovo argomento. TYPE Per esaminare una funzione di esempio, consulta la sezione Informazioni aggiuntive di questo modello.	Conoscenza di PostgreSQL PL/pgSQL
Compila la funzione PostgreSQL.	Compila la funzione in PostgreSQL. Se i nuovi campi del tipo di dati corrispondono agli argomenti della funzione o della procedura di origine, la funzione viene compilata correttamente.	Conoscenza di PostgreSQL PL/pgSQL

Operazione	Description	Competenze richieste
Esegui la funzione PostgreSQL.	Esegui la funzione PostgreSQL.	Conoscenza di PostgreSQL PL/pgSQL

risoluzione dei problemi

Problema	Soluzione
La funzione restituisce il seguente errore: ERRORE: errore di sintassi vicino a "" <statement>	Assicuratevi che tutte le istruzioni della funzione terminino con un punto e virgola (). `;`
La funzione restituisce il seguente errore: ERRORE: "" non è una variabile nota <variable>	Assicurati che la variabile utilizzata nel corpo della funzione sia elencata nella `DECLARE` sezione della funzione.

Risorse correlate

- Lavorare con Amazon Aurora PostgreSQL (Guida utente di Amazon Aurora per Aurora)
- CREATE TYPE (documentazione PostgreSQL)

Informazioni aggiuntive

Esempio di funzione PostgreSQL che include un argomento TYPE

```
CREATE OR REPLACE FUNCTION test_proc_new
(
    IN p_rec type_test_proc_args
)
RETURNS void
AS
$BODY$
BEGIN

    /*
```

```
    **************
    The body would contain code to process the input values.
    For our testing, we will display couple of values.
    ***************
    */
    RAISE NOTICE USING MESSAGE = CONCAT_WS('', 'p_acct_id: ', p_rec.p_acct_id);
    RAISE NOTICE USING MESSAGE = CONCAT_WS('', 'p_ord_id: ', p_rec.p_ord_id);
    RAISE NOTICE USING MESSAGE = CONCAT_WS('', 'p_ord_date: ', p_rec.p_ord_date);

END;
$BODY$
LANGUAGE plpgsql
COST 100;
```

Esegui la migrazione dei carichi di lavoro Redis su Redis Enterprise Cloud su AWS

Antony Prasad Thevaraj, Amazon Web Services

Srinivas Pendyala, Redis

Riepilogo

Questo modello illustra il processo di alto livello per la migrazione dei carichi di lavoro Redis su Redis Enterprise Cloud on Amazon Web Services (AWS). Descrive le fasi della migrazione, fornisce informazioni sulla selezione degli strumenti disponibili e illustra i vantaggi, gli svantaggi e le fasi di utilizzo di ciascuno strumento. Facoltativamente, se hai bisogno di ulteriore assistenza per la migrazione dei carichi di lavoro da Redis, puoi rivolgerti ai Redis Professional Services.

Se utilizzi Redis OSS o Redis Enterprise Software in locale, conosci bene il notevole sovraccarico amministrativo e la complessità operativa della manutenzione dei database Redis nel tuo data center. Migrando i carichi di lavoro sul cloud, puoi ridurre in modo significativo questo onere operativo e sfruttare Redis Enterprise Cloud, un'offerta di database as a service (S) completamente ospitata di Redis. DBaa Questa migrazione aiuta ad aumentare l'agilità aziendale, migliora l'affidabilità delle applicazioni e riduce i costi complessivi mentre accedi alle più recenti funzionalità di Redis Enterprise Cloud on AWS come disponibilità del 99,999%, semplicità architetturale e scalabilità.

Esistono potenziali applicazioni per Redis Enterprise Cloud nei settori dei servizi finanziari, della vendita al dettaglio, della sanità e dei giochi, nonché in casi d'uso che richiedono soluzioni per il rilevamento delle frodi, l'inventario in tempo reale, l'elaborazione dei reclami e la gestione delle sessioni. Puoi usare Redis Enterprise Cloud per connetterti alle tue risorse AWS, ad esempio, a un server di applicazioni in esecuzione su istanze Amazon Elastic Compute Cloud (Amazon EC2) o a un microservizio distribuito come servizio AWS Lambda.

Prerequisiti e limitazioni

Ipotesi

- Attualmente stai utilizzando un sistema di database locale che desideri migrare sul cloud.
- Hai identificato i requisiti di migrazione per i tuoi carichi di lavoro, tra cui:
  - Requisiti di coerenza dei dati
  - Requisiti dell'infrastruttura e dell'ambiente di sistema

  - Requisiti di mappatura e trasformazione dei dati
  - Requisiti per i test funzionali
  - Requisiti in materia di test
  - Requisiti di convalida
  - Strategia di cutover definita
- Avete valutato le tempistiche e le stime dei costi necessarie per la migrazione.
- I vostri requisiti prendono in considerazione l'ambito del lavoro e i sistemi e i database che avete identificato per far parte della migrazione.
- Avete identificato le parti interessate insieme ai loro ruoli e responsabilità in una matrice responsabile, responsabile, consultata e informata (RACI).
- Avete ricevuto l'accordo e le approvazioni necessari da tutte le parti interessate.

Costo

A seconda delle specifiche tecniche del database di origine esistente (ad esempio, dimensione della memoria, velocità effettiva e dimensione totale dei dati), un architetto di soluzioni Redis può dimensionare il sistema di destinazione su Redis Enterprise Cloud. Per informazioni generali sui prezzi, consulta la sezione Prezzi di Redis sul sito Web di Redis.

Persone e competenze

Il processo di migrazione prevede i seguenti ruoli e responsabilità.

Ruolo	Description	Competenze richieste
Architetto di soluzioni di migrazione	Un architetto tecnico esperto nella definizione, pianifica zione e implementazione di strategie di migrazione	Comprensione tecnica e applicativa dei sistemi di origine e destinazione; esperienza nella migrazione dei carichi di lavoro sul cloud
Architetto dei dati	Un architetto tecnico con una vasta esperienza nella definizione, implementazione e fornitura di soluzioni di dati per un'ampia varietà di database	Modellazione dei dati per dati strutturati e non strutturati, profonda comprensione ed esperienza nell'implementazio ne di database per un'azienda

Architetto di soluzioni Redis	Un architetto tecnico che può aiutare a progettare un cluster Redis di dimensioni ottimali per il caso d'uso appropriato	Esperienza nell'architettura e nell'implementazione di soluzioni Redis per un'ampia varietà di casi d'uso
Architetto di soluzioni cloud	Un architetto tecnico con una conoscenza più approfond ita delle soluzioni cloud, in particolare su AWS	Esperienza nell'architettura di soluzioni per il cloud; migrazione dei carichi di lavoro ed esperienza nella modernizz azione delle applicazioni
Architetto aziendale	Un architetto tecnico che ha una conoscenza completa del panorama tecnico dell'orga nizzazione, che ha una visione condivisa per la roadmap futura e che pratica e stabilisc e le migliori pratiche architett oniche standardizzate per tutti i team dell'organizzazione	Certificazioni di architett ura software come TOGAF, competenze di ingegneri a del software di base ed esperienza nell'architettura delle soluzioni e nell'arch itettura aziendale
IT o ingegnere DevOps	Un ingegnere responsab ile della creazione e della manutenzione dell'infrastruttur a, incluso il monitoraggio dell'infrastruttura per rilevare eventuali problemi, l'esecuzi one delle attività di manutenzi one e l'esecuzione degli aggiornamenti necessari.	Conoscenza approfondita di varie tecnologie, tra cui sistemi operativi, reti e cloud computing; familiarità con linguaggi di programma zione come Python, Bash e Ruby, nonché strumenti come Docker, Kubernetes e Ansible

Architecture

Opzioni di migrazione

Il diagramma seguente mostra le opzioni per la migrazione delle fonti di dati locali (basate su Redis o di altro tipo) su AWS. Mostra diversi strumenti di migrazione tra cui scegliere, come l'esportazione

di file Redis Database (RDB) su Amazon Simple Storage Service (Amazon S3), l'utilizzo della funzionalità di replica Redis o l'utilizzo di AWS DMS.

1. Fonti dati locali: database non basati su Redis, come MySQL, PostgreSQL, Oracle, SQL Server o MariadB.
2. Fonti di dati locali: database basati sul protocollo Redis come Redis OSS e Redis Enterprise Software.
3. Il modo più semplice per migrare i dati dai database basati su Redis consiste nell'esportare file RDB e importarli nel Redis Enterprise Cloud di destinazione su AWS.
4. In alternativa, puoi migrare i dati dall'origine alla destinazione utilizzando la funzionalità di replica () in Redis. `ReplicaOf`
5. Se i requisiti di migrazione dei dati includono la trasformazione dei dati, puoi utilizzare Redis Input/Output Tools (RIOT) per migrare i dati.
6. In alternativa, puoi utilizzare AWS Data Migration Service (AWS DMS) per migrare i dati da database basati su SQL.
7. È necessario utilizzare il peering del cloud privato virtuale (VPC) per AWS DMS per migrare correttamente i dati nel Redis Enterprise Cloud di destinazione su AWS.

Architettura Target

Il diagramma seguente mostra un'architettura di distribuzione tipica per Redis Enterprise Cloud su AWS e illustra come può essere utilizzata con i principali servizi AWS.

1. Puoi connetterti alle applicazioni aziendali supportate da Redis Enterprise Cloud on AWS.
2. Puoi eseguire applicazioni aziendali nel tuo account AWS, in un VPC all'interno di quell'account.
3. Puoi utilizzare gli endpoint del database Redis Enterprise Cloud per connetterti alle tue applicazioni. Gli esempi includono un server di applicazioni in esecuzione su EC2 istanze, un microservizio distribuito come servizio AWS Lambda, un'applicazione Amazon Elastic Container Service (Amazon ECS) o un'applicazione Amazon Elastic Kubernetes Service (Amazon EKS).
4. Le applicazioni aziendali in esecuzione nel tuo VPC richiedono una connessione peer VPC a Redis Enterprise Cloud VPC. Ciò consente alle applicazioni aziendali di connettersi in modo sicuro tramite endpoint privati.

5. Redis Enterprise Cloud on AWS è una piattaforma di database NoSQL in memoria distribuita come DBaa S su AWS ed è completamente gestita da Redis.
6. Redis Enterprise Cloud viene distribuito all'interno di un VPC in un account AWS standard creato da Redis.
7. Per motivi di sicurezza, Redis Enterprise Cloud è distribuito in una sottorete privata a cui è possibile accedere da endpoint privati e pubblici. Ti consigliamo di connettere le tue applicazioni client a Redis su endpoint privati. Se prevedi di utilizzare un endpoint pubblico, ti consigliamo vivamente di abilitare TLS per crittografare i dati tra le applicazioni client e Redis Enterprise Cloud.

La metodologia di migrazione Redis è in linea con la metodologia di migrazione AWS, illustrata in Mobilize your organization to accelerate migrazioni su larga scala sul sito Web AWS Prescriptive Guidance.

Automazione e scalabilità

Le attività di configurazione dell'ambiente per la migrazione possono essere automatizzate tramite modelli AWS Landing Zone e Infrastructure as Code (IaC) per l'automazione e la scalabilità. Questi sono discussi nella sezione Epics di questo modello.

Tools (Strumenti)

In base ai tuoi requisiti di migrazione dei dati, puoi scegliere tra una selezione di opzioni tecnologiche per migrare i tuoi dati su Redis Enterprise Cloud on AWS. La tabella seguente descrive e confronta questi strumenti.

Strumento	Description	Vantaggi	Svantaggi
Esportazione e importazione RDB	I dati vengono esportati dal database di origine (ad esempio, Redis OSS o Redis Enterprise Software) sotto forma di file RDB. Se il database viene fornito tramite un cluster Redis OSS, si esporta	• Semplice. • Funziona con qualsiasi soluzione basata su Redis in grado di esportare dati in formato RDB come sorgente (inclusi Redis OSS e Redis Enterprise Software).	• Non soddisfa i requisiti di trasformazione dei dati né supporta le unioni logiche di database. • Richiede molto tempo per set di dati di grandi dimensioni.

	ogni shard master in un RDB. Quindi importi tutti i file RDB in un unico passaggio . Se il database di origine è basato su un cluster OSS ma il database di destinazione non utilizza l'API del cluster OSS, è necessario modificare il codice sorgente dell'applicazione per utilizzare una libreria client Redis standard. I requisiti di trasformazione dei dati o le unioni logiche dei database richiedono un processo più complesso, illustrato nella sezione Unione logica dei database più avanti in questa tabella.	• Raggiunge la coerenza dei dati con un processo semplice.	• Nessun supporto per la migrazione delta può portare a tempi di inattività più lunghi.

Funzionalità di replica Redis (attiva-passiva)	È possibile replicare continuamente i dati da un database Redis OSS, Enterprise Software o Enterprise Cloud a un database Redis Enterpris e Cloud. Dopo la sincronizzazione iniziale, la funzional ità di replica Redis (`ReplicaOf`) esegue una migrazion e delta, il che significa che non si registrano quasi tempi di inattivit à delle applicazioni. La funzionalità di replica Redis è pensata per essere utilizzata in modo attivo-passivo. Si presume che la destinazione sia passiva e viene completamente risincronizzata (svuotata e sincroniz zata dal database di origine). Pertanto, il passaggio dalla sorgente alla destinazi one è leggermente più complicato.	• Supporta la replica continua (caricame nto iniziale dei dati seguito dai delta). • Quasi nessun tempo di inattività (dipende dal ritardo di replica). • Raggiunge la coerenza dei dati.	• È previsto che un solo sito sia attivo, quindi passare da un sito all'altro è più complicato. • Supporta un massimo di 32 master shard durante la migrazione da un cluster OSS.

	È possibile effettuare la replica da un cluster Redis OSS a un database Redis Enterprise Cloud in cluster standard specificando tutti gli shard master del cluster OSS come sorgenti. Tuttavia, la funzionalità di replica Redis consente un massimo di 32 database di origine.		
AWS DMS	Puoi utilizzare AWS DMS per migrare i dati da qualsiasi database di origine supportato a un data store Redis di destinazione con tempi di inattività minimi. Per ulteriori informazioni, consulta Using Redis come target per AWS DMS nella documentazione di AWS DMS.	• Supporta la migrazione di sorgenti dati NoSQL e SQL. • Funziona bene con altri servizi AWS. • Supporta casi d'uso di migrazione in tempo reale e acquisizione dei dati di modifica (CDC).	• I valori chiave Redis non possono contenere caratteri speciali come%. • Non supporta la migrazione di dati che contengono caratteri speciali nelle righe o nei nomi dei campi. • Non supporta la modalità LOB (Full Large Binary Object).

Unione logica di database	Requisiti speciali di unione dei database potrebbero richieder e una soluzione di migrazione dei dati personalizzata. Ad esempio, potreste avere quattro database logici (`SELECT 0..3`) in Redis OSS, ma potreste voler utilizzar e un singolo endpoint del database invece di spostare i dati su più database Redis Enterprise Cloud. Redis Enterprise non supporta database logici selezionabili, quindi è necessari o trasformare il modello di dati fisico del database di origine. Ad esempio, è possibile mappare ogni indice del database su un prefisso (`0`to `usrcmp`, 1 to e così via), quindi utilizzare uno script di migrazione o uno strumento di estrazion e, trasformazione e caricamento (ETL) per generare un file	• Controllo granulare sulla modellazione dei dati durante la migrazione al sistema di destinazi one utilizzando script personali zzati.	• Se si decide di non completare la migrazione, il rollback può essere molto impegnati vo, soprattutto se è necessario ripristinare i dati più recenti sui sistemi di origine. • Il costo di creazione può essere elevato se l'obiettivo è creare una soluzione unica per una migrazione una tantum. • I costi di manutenzi one per codice, infrastruttura, tempi di sviluppo e altre aree possono essere elevati se i requisiti di migrazione cambiano frequente mente.

	RDB, che è quindi possibile importare nel database di destinazione.

Inoltre, puoi utilizzare i seguenti strumenti e servizi di AWS.

Strumento di valutazione e scoperta:

- Migration Evaluator

Strumenti di migrazione di applicazioni e server:

- AWS Servizio della migrazione di applicazioni

Strumenti di migrazione del database:

- Strumento di conversione dello schema AWS (AWS SCT)
- AWS Database Migration Service (AWS DMS)

Strumenti per la migrazione dei dati:

- AWS Storage Gateway
- AWS DataSync
- AWS Direct Connect
- AWS Snowball
- Amazon Data Firehose

Soluzioni AWS Partner:

- Partner AWS con competenze per la migrazione

Epiche

Completa le attività di scoperta e valutazione

Operazione	Description	Competenze richieste
Identifica i carichi di lavoro.	Identifica i carichi di lavoro candidati adatti che desideri migrare. Considera quanto segue prima di scegliere un carico di lavoro per la migrazione: • Qual è il valore aziendale derivante dalla migrazione o dalla mancata migrazione di questo carico di lavoro? • Esiste un piano di emergenza se questo carico di lavoro non viene migrato correttamente al sistema di destinazione? Idealmente, scegli un carico di lavoro che abbia il massimo impatto aziendale con rischi minimi. Mantieni l'intero processo iterativo e migra in piccoli incrementi.	Architetto dei dati, campioni aziendali, sponsor di progetti di migrazione
Identifica le fonti e i requisiti dei dati; progetta un modello di dati.	Redis organizza un workshop per accelerare la scoperta e definire la pianificazione della migrazione per il progetto. Nell'ambito di questo workshop, i team Redis identificano le fonti di dati	Architetto di soluzioni Redis

Operazione	Description	Competenze richieste
	e i requisiti del modello di dati di origine e analizzano come questi possono essere rimodellati in Redis Enterprise Cloud. Il team di migrazione Redis (Professional Services) esegue un esercizio dettaglia to di progettazione del modello di dati con l'organizzazione. Nell'ambito di questo esercizio , il team Redis: • Identifica le strutture dati Redis target. • Definisce la strategia di mappatura dei dati. • Documenta l'approccio e le raccomandazioni relative alla migrazione. • Esamina e finalizza il modello di dati con le parti interessate.	

Operazione	Description	Competenze richieste
Identifica le caratteristiche del database di origine.	Identifica il prodotto Redis utilizzato negli ambienti di origine e di destinazione. Esempio: • Il database di origine è un database OSS Cluster, un database Redis autonomo o un database Redis Enterprise? • Il database di destinazione sarà un database standard Redis Enterprise o un database compatibile con OSS Cluster? • Quali sono le implicazioni relative al codice sorgente dell'applicazione?	Data architect
Raccogli gli SLA attuali del sistema e altre metriche di dimensionamento.	Determina gli attuali accordi sui livelli di servizio (SLAs) espressi in termini di velocità effettiva (operazioni al secondo), latenza, dimensione complessiva della memoria per database e requisiti di alta disponibilità (HA).	Data architect

Operazione	Description	Competenze richieste
Identifica le caratteristiche del sistema di destinazione.	Determina le risposte a queste domande: • Quanti dati devono essere migrati? • Quanto tempo occorre per migrare una determinata quantità di dati? • Quali sono i requisiti relativi ai tempi di inattività per la migrazione? È accettabile che il servizio o l'applica zione non siano disponibili per un periodo specifico? In caso affermativo, per quanto tempo? • Quanto devono essere coerenti i dati migrati? Il database di destinazione può trovarsi in uno stato leggermente incoerente (obsoleto)? • I dati devono essere trasformati prima di essere caricati nel database di destinazione? (Ad esempio, potresti voler convertir e gli indici DB seleziona bili in prefissi prima della migrazione.) • Il database di origine è raggiungibile dall'host del database di destinazione (ad esempio, da un VPC	Architetto dei dati, architetto delle soluzioni Redis (opzional e)

Operazione	Description	Competenze richieste
	peer o da un endpoint pubblico utilizzando la crittografia)? • Completa un esercizio di dimensionamento dei dati e di dimensionamento del cluster Redis con un architetto tecnico Redis. • Identifica i requisiti di rete, i requisiti di infrastruttura, le versioni del software e le licenze software e procurati tutti i componenti prima della migrazione. • Esistono problemi di sicurezza associati al trasferimento di questi dati?	
Identifica le dipendenze.	Identifica le dipendenze a monte e a valle del sistema corrente da migrare. Assicurati che il processo di migrazione sia in linea con le altre migrazioni di sistemi dipendenti. Ad esempio, se hai intenzione di migrare altre applicazioni aziendali dall'ambiente locale al cloud AWS, identifica queste applicazioni e allineale in base agli obiettivi del progetto, alle tempistiche e alle parti interessate.	Architetto dei dati, architetto aziendale

Operazione	Description	Competenze richieste
Identifica gli strumenti di migrazione.	A seconda dei requisiti di migrazione dei dati (come i dati di origine o i requisiti relativi ai tempi di inattività), puoi utilizzare uno qualsiasi degli strumenti descritti in precedenza nella sezione Strumenti. Inoltre, puoi utilizzare: • Replica bidirezionale (attiva-attiva) utilizzando la distribuzione CRDB. • export/import Script personalizzati (ad esempio, utilizzando i comandi). `DUMP/RESTORE` • export/import Strumenti aggiuntivi e strumenti di supporto come RIOT o ECstats2ETL. • Strumenti IaC come i modelli Terraform o CloudFormation AWS.	Architetto di soluzioni di migrazione, architetto di soluzioni Redis
Crea un piano di emergenza.	Stabilisci un piano di emergenza per il rollback, in caso di problemi durante la migrazione.	Gestione del progetto, team tecnici, incluso l'architetto

Completa le attività di sicurezza e conformità

Operazione	Description	Competenze richieste
Proteggi la console di amministrazione Redis.	Per proteggere la console di amministrazione, segui le istruzioni nella documenta zione Redis.	Amministratore dell'infr astruttura IT
Proteggi il database Redis.	Consulta le seguenti pagine nella documentazione di Redis per: • Definire il controllo degli accessi basato sui ruoli. • Definisci la sicurezza della rete. • Abilita TLS.	
Redis Cloud sicuro. APIs	Quando abiliti l'API, puoi gestire le chiavi API per tutti i proprietari del tuo account Redis Cloud. Per una panoramica delle funzionalità di sicurezza dell'API, consulta la documentazione sull'aute nticazione dell'API sul sito Web Redis.	Amministratore dell'infr astruttura IT

Configura il nuovo ambiente

Operazione	Description	Competenze richieste
Configura un nuovo ambiente su AWS.	Questa attività include:	IT o ingegnere DevOps

Operazione	Description	Competenze richieste

	• Attività di configurazione di AWS Landing Zone. La landing zone supporta:
	  • Implementazioni con più account
	  • Base minima di sicurezza
	  • Modo automatizzato per fornire nuovi account con una linea di base di sicurezza e prerequisiti ISV (rete, configurazione di sicurezza e così via)
	  • Notifiche, registrazione centralizzata e monitoraggio
	• Attività di configurazione del software ISV. Ciò include le configurazioni che devono essere incluse nella migrazione, come le impostazioni e le modifiche del prodotto e del carico di lavoro.
	• Attività IaC come la configurazione o la personalizzazione di modelli CloudFormation AWS o Terraform.	

Operazione	Description	Competenze richieste
Implementa l'architettura di migrazione.	1. Configura Redis Enterprise Cloud su AWS. 2. Installa strumenti di migrazione come RIOT o AWS DMS. Consulta la sezione Strumenti per un elenco degli strumenti disponibili. 3. Stabilisci la connettività tra i livelli di applicazione, migrazione e database. 4. Crea un carico di lavoro di esempio che possa fluire attraverso ogni livello e migra un piccolo set di dati di esempio. Ora sei pronto per eseguire le pipeline di migrazione dei dati effettive e testarle.	IT o DevOps ingegnere

Configura la rete

Operazione	Description	Competenze richieste
Stabilire la connettività.	Stabilisci la connettività tra l'infrastruttura locale e le risorse del cloud AWS. Utilizza gruppi di sicurezza, AWS Direct Connect e altre risorse per ottenere questa funzionalità. Per ulteriori informazioni, consulta Connect Your Data	IT o DevOps ingegnere

Operazione	Description	Competenze richieste
	Center to AWS sul sito Web di AWS.	
Configura il peering VPC.	Stabilisci il peering VPC tra le istanze VPCs che eseguono applicazioni aziendali (o le EC2 istanze che eseguono strumenti di migrazione o il server di replica AWS DMS) e il VPC che esegue Redis Enterprise Cloud. Per istruzioni, consulta Introduzione ad Amazon VPC nella documentazione di Amazon VPC e Abilita il peering VPC nella documentazione Redis.	IT o ingegnere DevOps

Migrazione dei dati

Operazione	Description	Competenze richieste
Scegli uno strumento di migrazione dei dati.	Consulta la tabella nella sezione Strumenti per vedere le descrizioni, i vantaggi e gli svantaggi di questi strumenti: • Esportazione e importazione RDS • Funzionalità di replica Redis () `ReplicaOf` • AWS DMS • Unione logica di database	Architetto di soluzioni di migrazione

Operazione	Description	Competenze richieste
	Le righe seguenti descrivono le attività di migrazione dei dati associate a ogni strumento.	
Opzione 1: utilizza l'esportazione e l'importazione da RDB.	1. Disconnetti la fonte: interrompi il traffico sul database di origine (ad esempio, disconnettendo le applicazioni aziendali). 2. Esporta: esporta i dati del database di origine come file RDB. 3. Fase: carica i dati in una posizione accessibile alle istanze Redis Enterprise Cloud su AWS (ad esempio, puoi caricarli su un bucket S3 o un server FTP). 4. Importa: importa i file RDB (elencandoli tutti in un unico passaggio di importazione) nel database di destinazione Redis Enterprise Cloud. 5. Cut over: passa al database di destinazione (ad esempio, connettendo l'applicazione, connettiti ad esso). Per ulteriori informazioni, consulta la documentazione di Redis.	Architetto di soluzioni di migrazione, architetto di soluzioni Redis

Operazione	Description	Competenze richieste
Opzione 2: utilizza la funzionalità di replica Redis (attiva-passiva).	1. Connect database: Stabilisci un `ReplicaOf` collegamento tra i database di origine e di destinazione. 2. Esegui una sincronizzazione iniziale: attendi il completamento della sincronizzazione iniziale tra i database di origine e di destinazione. 3. Disconnetti la sorgente: interrompi il traffico sul database di origine (ad esempio, disconnettendo l'applicazione). 4. Esegui la replica delta: attendi che il delta venga replicato sul database di destinazione. 5. Cut over: passa al database di destinazione (ad esempio, collegando l'applicazione ad esso). 6. Elimina: rimuove il `ReplicaOf` collegamento tra il database di origine e quello di destinazione. Per ulteriori informazioni, consulta la documentazione di Redis.	Architetto di soluzioni di migrazione, architetto di soluzioni Redis

Operazione	Description	Competenze richieste
Opzione 3: usa AWS DMS.	1. Configura un'istanza di replica AWS DMS: questa istanza esegue tutti i processi di migrazione. Per istruzioni: Utilizzo di un'istanza di replica AWS DMS nella documentazione di AWS DMS. 2. Definisci il database di origine: definisci l'endpoint di origine. Verifica la connettività tra l'endpoint di origine e il server di replica AWS DMS. Per istruzioni: creazione di endpoint di origine e destinazione nella documentazione di AWS DMS. 3. Configura il database di destinazione: configura Redis Enterprise Cloud su AWS e configura il database su cui migrare. 4. Definisci il database di destinazione: definisci l'endpoint di destinazione. Assicurati che il peering VPC sia stabilito tra il VPC su cui è in esecuzione AWS DMS e il VPC che ospita Redis Enterprise Cloud su AWS. Verifica la connettività tra il server di replica	Architetto di soluzioni di migrazione, architetto di soluzioni Redis

Operazione	Description	Competenze richieste
	AWS DMS e il database di destinazione. 5. Crea un'attività AWS DMS: crea un'attività o un insieme di attività per definire le tabelle e i processi di replica che desideri utilizzare per migrare i dati. Per istruzioni: Utilizzo delle attività di AWS DMS nella documentazione di AWS DMS. 6. Migrazione: migra i dati eseguendo il task AWS DMS. 7. Cut over: passa al database di destinazione (ad esempio, collegando l'applicazione ad esso).	
Opzione 4: utilizzare l'unione logica dei database.	Questa opzione prevede l'utilizzo di uno script di migrazione o di uno strumento ETL in grado di trasformare il modello di dati fisico del database di origine e generare un file RDB. Redis Professio nal Services può aiutarti in questa fase, se necessario.	Architetto di soluzioni di migrazione, architetto di soluzioni Redis

Migra la tua applicazione

Operazione	Description	Competenze richieste
Allinea le tempistiche e gli obiettivi di gestione del progetto.	Allinea gli obiettivi, le tappe fondamentali e le tempistiche del progetto di migrazione del livello applicativo con quelli del progetto di migrazione dei dati Redis.	Gestione progettuale
Allinea le attività di test.	Dopo la migrazione e la modernizzazione del livello applicativo nel cloud AWS, indirizza il livello applicativo al Redis Enterprise Cloud on AWS appena migrato per i test.	Test in corso

Test

Operazione	Description	Competenze richieste
Implementa piani di test.	Esegui le routine di migrazione dei dati e gli script sviluppati durante la fase di implementazione in un ambiente di test, in base ai requisiti di test, presso il tuo sito.	Test in corso
Verifica la qualità dei dati.	Verifica la qualità dei dati dopo la migrazione dei dati.	Test in corso
Funzionalità di test.	Verifica le interrogazioni sui dati e il livello di applicazione per assicurarti che l'applica	Test in corso

Operazione	Description	Competenze richieste
	zione funzioni allo stesso livello del sistema di origine.	

Tagliare

Operazione	Description	Competenze richieste
Prendi la decisione finale.	Una volta completati tutti i test a livello di applicazione e database, il team dirigenziale esecutivo e le parti interessa te prendono la decisione finale in merito all'opportunità di passare al nuovo ambiente su AWS sulla base dei risultati finali confermati dai team di test.	Gestione dei progetti, Business champions
Passa al cloud AWS.	Una volta confermato che tutto è a posto, indirizza il livello applicativo ai dati appena migrati e indirizza i client al nuovo livello applicativo in esecuzione basato sul nuovo sistema Redis Enterprise Cloud su AWS.	Ingegnere IT o DevOps tecnico, architetto dei dati, architetto delle soluzioni di migrazione, architetto delle soluzioni Redis

Risorse correlate

Risorse Redis

- Documentazione Redis Enterprise Cloud
- Strumento RIOT (repository) GitHub
- Terraform Provider (scarica)

Risorse AWS

- Migrazioni demo
- Soluzioni per partner AWS
- Documentazione
- Post di blog
- Libri bianchi
- Tutorial e video
- Migrazione al cloud AWS
- Prontuario AWS

Informazioni aggiuntive

Per i requisiti di sicurezza standard per la migrazione dei carichi di lavoro Redis sul cloud AWS, consulta le Best Practices for Security, Identity and Compliance sul sito Web AWS e il Redis Trust Center sul sito Web Redis.

Esegui la migrazione da SAP HANA ad AWS utilizzando SAP HSR con lo stesso nome host

Pradeep Puliyampatta, Amazon Web Services

Riepilogo

Le migrazioni SAP HANA verso Amazon Web Services (AWS) possono essere eseguite utilizzando diverse opzioni, tra cui backup e ripristino, esportazione e importazione e SAP HANA System Replication (HSR). La selezione di una particolare opzione dipende dalla connettività di rete tra i database SAP HANA di origine e di destinazione, dalla dimensione del database di origine, da considerazioni relative ai tempi di inattività e da altri fattori.

L'opzione SAP HSR per la migrazione dei carichi di lavoro SAP HANA su AWS funziona bene quando è presente una rete stabile tra il sistema di origine e quello di destinazione e l'intero database (snapshot di replica SAP HANA DB) può essere replicato completamente entro 1 giorno, come stabilito da SAP per i requisiti di throughput di rete per SAP HSR. I requisiti di downtime con questo approccio si limitano all'esecuzione dell'acquisizione sull' AWS ambiente di destinazione, al backup di SAP HANA DB e alle attività successive alla migrazione.

SAP HSR supporta l'uso di diversi nomi host (nomi host mappati su diversi indirizzi IP) per il traffico di replica tra i sistemi primario, o di origine, e secondario o di destinazione. È possibile farlo definendo quei set specifici di nomi host nella sezione in. `[system_replication_hostname_resolution]` `global.ini` In questa sezione, tutti gli host dei siti primario e secondario devono essere definiti su ciascun host. Per i passaggi di configurazione dettagliati, consulta la documentazione SAP.

Un aspetto fondamentale di questa configurazione è che i nomi host nel sistema primario devono essere diversi dai nomi host nel sistema secondario. In caso contrario, è possibile osservare i seguenti errori.

- `"each site must have a unique set of logical hostnames"`
- `"remoteHost does not match with any host of the source site. All hosts of source and target site must be able to resolve all hostnames of both sites correctly"`

Tuttavia, il numero di passaggi successivi alla migrazione può essere ridotto utilizzando lo stesso nome host SAP HANA DB nell'ambiente di destinazione. AWS

Questo modello fornisce una soluzione alternativa per utilizzare lo stesso nome host negli ambienti di origine e di destinazione quando si utilizza l'opzione SAP HSR. Con questo modello, è possibile utilizzare l'opzione SAP HANA Hostname Rename. Si assegna un nome host temporaneo al database SAP HANA di destinazione per facilitare l'unicità del nome host per SAP HSR. Dopo che la migrazione ha completato la fase cardine dell'acquisizione sull'ambiente SAP HANA di destinazione, è possibile ripristinare il nome host del sistema di destinazione nel nome host del sistema di origine.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Un cloud privato virtuale (VPC) con un endpoint o un router di rete privata virtuale (VPN).
- AWS Client VPN o AWS Direct Connect configurato per trasferire file dall'origine alla destinazione.
- Database SAP HANA sia nell'ambiente di origine che in quello di destinazione. Il livello di patch SAP HANA DB di destinazione deve essere uguale o superiore al livello di patch SAP HANA DB di origine, all'interno della stessa edizione della piattaforma SAP HANA. Ad esempio, la replica non può essere configurata tra i sistemi HANA 1.0 e HANA 2.0. Per ulteriori informazioni, consulta la domanda 15 in SAP Nota: 1999880 — Domande frequenti: SAP HANA System Replication.
- Server applicativi SAP nell'ambiente di destinazione.
- Volumi Amazon Elastic Block Store (Amazon EBS) nell'ambiente di destinazione.

Limitazioni

Il seguente elenco di documenti SAP copre i problemi noti relativi a questa soluzione alternativa, inclusi i vincoli relativi al tiering dinamico su più livelli e alle migrazioni con scalabilità orizzontale di SAP HANA:

- 2956397 — Ridenominazione del sistema di database SAP HANA non riuscita
- 2222694 - Quando si tenta di rinominare il sistema HANA, viene visualizzato il seguente errore «I file di origine non sono di proprietà dell'utente sidadm originale (uid = xxxx)»
- 2607227 — hdblcm: register_rename_system: ridenominazione dell'istanza SAP HANA non riuscita
- 2630562 — HANA Hostname Rename non riuscita e HANA non si avvia
- 2935639 — sr_register non utilizza il nome host specificato in system_replication_hostname_resolution nella sezione global.ini

- 2710211 — Errore: il sistema di origine e il sistema di destinazione hanno nomi host logici sovrapposti
- 2693441 - Impossibile rinominare un sistema SAP HANA a causa di un errore
- 2519672 - HANA Primary e Secondary hanno dati e chiavi PKI SSFS di sistema diversi o non possono essere controllati
- 2457129 — La ridenominazione dell'host del sistema SAP HANA non è consentita quando il tiering dinamico fa parte del panorama
- 2473002 — Utilizzo di HANA System Replication per migrare un sistema con scalabilità orizzontale (SAP non prevede alcuna restrizione nell'utilizzo di questo approccio di ridenominazione degli host per i sistemi SAP HANA con scalabilità orizzontale. Tuttavia, la procedura deve essere ripetuta su ogni singolo host. A questo approccio si applicano anche altre limitazioni della migrazione con scalabilità orizzontale.)

Versioni del prodotto

- Questa soluzione si applica alle edizioni 1.0 e 2.0 della piattaforma SAP HANA DB.

Architecture

Configurazione del codice sorgente

Un database SAP HANA è installato nell'ambiente di origine. Tutte le connessioni all'application server SAP e le interfacce DB utilizzano lo stesso nome host per le connessioni client. Il diagramma seguente mostra l'esempio del nome host di origine `hdbhost` e l'indirizzo IP corrispondente.

Configurazione del bersaglio

L'ambiente Cloud AWS di destinazione utilizza lo stesso nome host per eseguire un database SAP HANA. L'ambiente di destinazione su AWS include quanto segue:

- Database SAP HANA
- Server di applicazioni SAP
- Volumi EBS

Configurazione intermedia

Nel diagramma seguente, il nome host nell'ambiente di AWS destinazione viene temporaneamente rinominato in `temp-host` modo che i nomi host sull'origine e sulla destinazione siano univoci. Dopo che la migrazione ha completato la fase cardine dell'acquisizione nell'ambiente di destinazione, il nome host virtuale del sistema di destinazione viene rinominato utilizzando il nome originale,. `hdbhost`

La configurazione intermedia include una delle seguenti opzioni:

- AWS Client VPN con un endpoint Client VPN
- Direct Connect connessione a un router

I server delle applicazioni SAP nell'ambiente di AWS destinazione possono essere installati prima della configurazione della replica o dopo l'acquisizione. Tuttavia, l'installazione degli application server prima della configurazione della replica può contribuire a ridurre i tempi di inattività durante l'installazione, la configurazione dell'alta disponibilità e i backup.

Tools (Strumenti)

Servizi AWS

- AWS Client VPNè un servizio VPN gestito basato su client che consente di accedere in modo sicuro alle AWS risorse e alle risorse della rete locale.
- AWS Direct Connectcollega la rete interna a una Direct Connect posizione tramite un cavo Ethernet standard in fibra ottica. Con questa connessione, è possibile creare interfacce virtuali direttamente verso il pubblico Servizi AWS, evitando i provider di servizi Internet presenti nel percorso di rete.
- Amazon Elastic Block Store (Amazon EBS) fornisce volumi di storage a livello di blocco da utilizzare con istanze Amazon Elastic Compute Cloud (EC2Amazon). Il comportamento dei volumi EBS è simile a quello dei dispositivi a blocchi non formattati e non elaborati. Puoi montare questi volumi come dispositivi sulle istanze.

Altri strumenti

- Server di applicazioni SAP: gli application server SAP forniscono ai programmatori un modo per esprimere la logica di business. L'application server SAP esegue l'elaborazione dei dati in base alla logica aziendale. I dati effettivi vengono archiviati in un database, che è un componente separato.
- SAP HANA Cockpit e SAP HANA Studio — Sia SAP HANA cockpit che SAP HANA Studio forniscono un'interfaccia amministrativa per il database SAP HANA. In SAP HANA Studio, la console di amministrazione SAP HANA è la visualizzazione del sistema che fornisce i contenuti pertinenti per l'amministrazione del database SAP HANA.
- Replicazione del sistema SAP HANA — SAP HANA System Replication (SAP HSR) è la procedura standard fornita da SAP per la replica dei database SAP HANA. Gli eseguibili richiesti per SAP HSR fanno parte del kernel del server SAP HANA stesso.

Epiche

Prepara gli ambienti di origine e di destinazione

Operazione	Description	Competenze richieste
Installa e configura i database SAP HANA.	Negli ambienti di origine e di destinazione, assicurati che il database SAP HANA sia installato e configurato secondo le best practice di SAP HANA. Per ulteriori informazioni, consulta SAP HANA on. AWS	Amministrazione SAP Basis
Mappa l'indirizzo IP.	Nell'ambiente di destinazione, assicuratevi che il nome host temporaneo sia assegnato a un indirizzo IP interno. 1. Assegna un IPv4 indirizzo secondario all' EC2 istanza nella Console di gestione AWS accedendo a EC2, Instance, Actions, Networking, Manage IP	Amministrazione AWS

Operazione	Description	Competenze richieste
	address, Assign new IP address. 2. Per assegnare lo stesso indirizzo all'adattatore di EC2 rete (NIC), dal sistema operativo, come utente root, esegui il comando`ip addr add <IP>/32 dev eth0`, sostituendolo `<IP>` con l'indirizzo IP del passaggio 1.	
Risolvi i nomi host di destinazione.	Sul DB SAP HANA secondario, verifica che entrambi i nomi host (`hdbhostetemp-host`) siano stati risolti per le reti di replica SAP HANA aggiornando i nomi host pertinenti nel file. `/etc/hosts`	Amministrazione Linux
Esegui il backup dei database SAP HANA di origine e di destinazione.	Utilizza SAP HANA Studio o il cockpit SAP HANA per eseguire backup sui database SAP HANA.	Amministrazione SAP Basis

Operazione	Description	Competenze richieste
Certificati PKI del sistema Exchange.	(Si applica solo a SAP HANA 2.0 e versioni successive) I certificati Exchange vengono scambiati nell'archivio sicuro dell'infrastruttura a chiave pubblica (PKI) del sistema nell'archivio del file system (SSFS) tra i database primari e secondari. Per ulteriori informazioni, vedere SAP Note 2369981 — Passaggi di configurazione richiesti per l'autenticazione con SAP HANA System Replication.	Amministrazione di SAP Basis

Rinomina il database SAP HANA di destinazione

Operazione	Description	Competenze richieste
Interrompi le connessioni dei client di destinazione.	Nell'ambiente di destinazione, chiudi i server delle applicazioni SAP e le altre connessioni client.	Amministrazione SAP Basis

Rinomina il database SAP HANA di destinazione con il nome host temporaneo.	1. Come utente root, rinomina il nome host SAP HANA DB di destinazione con il nome host temporaneo utilizzando resident. `hdblcm`
	`root $> cd /hana/shared/<SID/hdblcm`
	`root $> ./hdblcm`
		Amministrazione SAP Basis

Operazione	Description	Competenze richieste
	2. Scegli l'opzione. `9 | rename_system | Rename the SAP HANA Database System` 3. Fornisci il nuovo nome: `temp-host` . 4. Puoi convalidare altre opzioni secondo necessità. Tuttavia, assicuratevi di non confondere la ridenomin azione dell'host con una modifica del SID (Nota SAP 2598814 — hdblcm: la ridenominazione SID fallisce). Lo stop and start di SAP HANA DB sarà controllato da. `hdblcm`	
Assegna reti di replica.	Nel `global.ini` file del sistema di origine, sotto l'`[system_replicatio n_hostname_resolut ion]` intestazione, fornisci i dettagli della rete di replica di origine e di destinazione. Copiate quindi le voci nel `global.ini` file sul sistema di destinazione.	Amministrazione SAP Basis

Operazione	Description	Competenze richieste
Abilita la replica sul sistema primario.	Per abilitare la replica sul DB SAP HANA di origine, esegui il comando seguente. `hdbnsutil -sr_enable --name=siteA`	Amministrazione SAP Basis

Registrare il database SAP HANA di destinazione come sistema secondario.	Per registrare il database SAP HANA di destinazione come sistema secondario di origine per SAP HSR, scegli la replica asincrona.
	`(sid)adm $> HDB stop`
	`(sid)adm $> hdbnsutil -sr_register –name=siteB –remotehost=hdbhost /`
	`--remoteInstance=00 –replicationMode=async –operationMode=logreplay`
	`(sid)adm $> HDB start`
	In alternativa, puoi scegliere l'opzione di registrazione. `–online` In tal caso, non è necessario arrestare e avviare il database SAP HANA.
		Amministrazione SAP Basis

Operazione	Description	Competenze richieste

Convalida la sincronizzazione.	Sul DB SAP HANA di origine, verifica che tutti i log siano applicati al sistema di destinazione (poiché si tratta di una replica asincrona).
	
	Per verificare la replica, esegui i seguenti comandi sull'origine.
	
	`(sid)adm $> cdpy`
	`(sidadm $> python systemReplicationStatus.py`
		Amministrazione SAP Basis

Chiudi l'applicazione SAP di origine e SAP HANA DB.	Durante il cutover della migrazione, esegui uno spegnimento del sistema di origine (l'applicazione SAP e il database SAP HANA).	Amministrazione di SAP Basis
Effettua un'acquisizione sull'obiettivo.	Per eseguire un'acquisizione sulla destinazione su AWS, esegui il comando`hdbnsutil -sr_takeover` .	Amministrazione SAP Basis

Operazione	Description	Competenze richieste
Sul DB SAP HANA di destinazione, disattiva la replica.	Per cancellare i metadati di replica, interrompi la replica sul sistema di destinazione eseguendo il comando. `hdbnsutil -sr_disable` **Note** Ciò è conforme alla nota SAP 2693441: impossibile rinominare un sistema SAP HANA a causa di un errore.	Amministrazione SAP Basis
Esegui il backup del database SAP HANA di destinazione.	Una volta completata con successo l'acquisizione, consigliamo di eseguire un backup completo di SAP HANA DB.	Amministrazione SAP Basis

Ripristina il nome host originale nel sistema di destinazione

Operazione	Description	Competenze richieste

Ripristina il nome host SAP HANA DB di destinazione all'originale.	1. Per ripristinare il nome host SAP HANA DB di destinazione al nome host virtuale originale, usa resident. `hdblcm`
	
	`root $> cd /hana/shared/<SID>/hdblcm`
	`root $> ./hdblcm`
		Amministrazione SAP Basis

Operazione	Description	Competenze richieste
	2. Scegli `9 | rename_system | Rename the SAP HANA Database System` l'opzione. 3. Fornisci il nuovo nome:`hdbhost`. Puoi convalidare altre opzioni secondo necessità. Tuttavia, assicuratevi di non confondere la ridenominazione dell'host con una modifica del SID (Nota SAP 2598814 — hdblcm: la ridenominazione SID fallisce).	
Regola hdbuserstore.	Adatta i `hdbuserstore` dettagli puntando ai dettagli della fonte. `schema/user` Per i passaggi dettagliati, consulta la documentazione SAP. Per convalidare questo passaggio, esegui il comando. `R3trans -d` Il risultato dovrebbe riflettere una connessione riuscita al database SAP HANA.	Amministrazione SAP Basis
Avvia le connessioni client.	Nell'ambiente di destinazione, avvia i server delle applicazioni SAP e altre connessioni client.	Amministrazione SAP Basis

Risorse correlate

Riferimenti SAP

I riferimenti alla documentazione SAP vengono aggiornati frequentemente da SAP. Per rimanere aggiornato, consulta la nota SAP 2407186 — Guide pratiche e white paper per SAP HANA High Availability.

Note SAP aggiuntive

- 2550327 — Come rinominare un sistema SAP HANA
- 1999880 — Domande frequenti: replica del sistema SAP HANA
- 2078425 — Nota sulla risoluzione dei problemi per lo strumento di gestione del ciclo di vita della piattaforma SAP HANA hdblcm
- 2592227 — Modifica del suffisso FQDN nei sistemi HANA
- 2048681 — Esecuzione di attività di amministrazione della gestione del ciclo di vita della piattaforma SAP HANA su sistemi con più host senza SSH o credenziali root

Documenti SAP

- Connessione di rete per la replica del sistema
- Risoluzione dei nomi host per la replica del sistema

AWS riferimenti

- Migrazione di SAP HANA da altre piattaforme a AWS

Informazioni aggiuntive

Le modifiche eseguite da nell'`hdblcm`ambito dell'attività di ridenominazione del nome host vengono consolidate nel seguente registro dettagliato.

Esegui la migrazione del gruppo di disponibilità Microsoft SQL Server Always On utilizzando AWS Application Migration Service

Sreenivas Nettem, Bharath Kumar Pammi Ramesh, Anantharaman Seshadri e Gireesh Sreekantan, Amazon Web Services

Riepilogo

AWS Application Migration Service (AWS MGN) è lo strumento preferito per il rehosting degli ambienti esistenti in, che consente ai clienti di abbandonare i data center locali. Cloud AWS Questo modello descrive il processo di utilizzo di AWS MGN per migrare i cluster Windows con i gruppi di disponibilità Always On di Microsoft SQL Server.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Un ruolo AWS Identity and Access Management (IAM) per l'orchestrazione di AWS MGN.
- Accesso ai server di database di origine (gruppo di disponibilità SQL Server Always On).
- Active Directory in una AWS landing zone per conservare i nomi DNS.
- Una sottorete temporanea con comunicazione di rete chiusa con Active Directory.
- Le sottoreti di destinazione sono in grado di comunicare con Active Directory.
- Due indirizzi IP riservati per il cluster Windows in una sottorete di destinazione (uno in ogni zona di disponibilità).
- Due indirizzi IP riservati per il listener SQL Always On in una sottorete di destinazione (uno in ogni zona di disponibilità).

Versioni del prodotto

- Windows Server 2012 o versioni successive
- SQL Server 2012 o versione successiva

Architecture

Stack tecnologico di origine

Cluster Microsoft Windows (macchina fisica o virtuale locale) Gruppo di disponibilità Microsoft SQL Server Always On

Stack tecnologico Target

Istanza Amazon EC2 Windows

Architettura Target

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- AWS Application Migration Serviceti aiuta a riospitare (lift and shift) le applicazioni Cloud AWS senza modifiche e con tempi di inattività minimi.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.

Altri strumenti

- Microsoft SQL Server Management Studio (SSMS) è uno strumento per la gestione di SQL Server, che include l'accesso, la configurazione e l'amministrazione dei componenti di SQL Server.

Best practice

Per AWS MGN, consulta Best practice for. AWS Application Migration Service

Epiche

Prepara l'account di destinazione

Operazione	Description	Competenze richieste
Inizializza AWS MGN.	Inizializza AWS MGN nella destinazione. Regione AWS	Amministratore cloud

Operazione	Description	Competenze richieste
	Questo crea i ruoli e le politiche IAM richiesti. Per ulteriori informazioni, consulta Inizializzazione del servizio di migrazione delle applicazioni con la console.	
Crea modelli di replica e avvia.	Configura i modelli di replica e avvio da utilizzare con AWS MGN. Per ulteriori informazioni, vedete Configurazione dei modelli nella documentazione. AWS	Amministratore cloud
Consenti le porte di comunicazione.	Per abilitare le comunicazioni di rete per AWS MGN, consenti il traffico sulle porte TCP 443 e 1500. Per ulteriori informazioni, consulta i requisiti di rete per il servizio di migrazione delle applicazioni nella documentazione. AWS	Amministratore cloud, amministratore di rete

Preparare il server di origine

Operazione	Description	Competenze richieste
Verifica i prerequisiti AWS MGN.	Verificate che i server di origine soddisfino i prerequisiti per l'installazione dell'agente AWS MGN. Per ulteriori informazioni, consultate Requisiti di installazione nella AWS documentazione.	Tecnico di migrazione

Operazione	Description	Competenze richieste
Installa l'agente AWS MGN.	Installa l'agente AWS MGN sui server di origine. Durante l'installazione, scegliete Regione AWS dove migrare i server. Dopo l'installazione, l'agente comunica con il servizio e avvia la replica. Per ulteriori informazioni, vedere Installazione dell'agente di AWS replica sui server Windows.	Ingegnere della migrazione
Controlla lo stato del server di origine.	Nella console AWS MGN, controllate lo stato dei server di origine. All'inizio della replica, sui server viene visualizzato il messaggio Pronto per il test. In caso di errori, vedere Risoluzione degli errori di comunicazione nella documentazione di AWS MGN.	Amministratore del cloud, tecnico addetto alla migrazione

Operazione	Description	Competenze richieste
Ottimizza le impostazioni di replica.	I cluster SQL Always On utilizzano una replica I/O sincrona elevata dal server primario al server secondari o. Per ottimizzare la replica ed evitare ritardi, utilizza un server di replica dedicato per ogni server SQL Always On. Se il database è superiore a 5 TB, valuta la possibilità di scegliere un'istanza del server di replica di dimension i maggiori, ad esempio m5.large, anziché la t3.small predefinita.	Amministratore del cloud, ingegnere addetto alla migrazione
Aggiorna il modello di lancio.	Aggiorna le impostazioni di avvio e scegli le sottoreti per i server SQL Always On. I server del cluster SQL Always On sono distribuiti su diversi server AWS Zone di disponibi lità per garantire un'elevata disponibilità.	Ingegnere addetto alla migrazione, responsabile della migrazione

Operazione	Description	Competenze richieste
Aggiorna le impostazioni di avvio.	In base ai requisiti di dimensioni e prestazioni, aggiorna il tipo di istanza e input/output le operazioni al secondo (IOPS) nelle impostazioni di avvio. (Facoltativo) Scegli un'interfaccia elastic network esistente nelle impostazioni di avvio.	Ingegnere addetto alla migrazione, responsabile della migrazione

Cutover di prova

Operazione	Description	Competenze richieste
Verifica i server di origine.	Nella console AWS MGN, verificate che lo stato del server di origine sia Pronto per il test.	Amministratore del cloud, ingegnere addetto alla migrazione
Avvia le istanze di test.	1. Avvia le istanze di test e verifica nella EC2 console Amazon che i controlli automatici abbiano esito positivo. 2. Seleziona e avvia le istanze di test per il server di controllo. 3. Verifica di poter accedere ai server tramite la console AWS MGN. 4. Seleziona i server del cluster SQL Always On e	Amministratore cloud, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	avvia contemporaneamente le istanze di test.	
Test di connettività e integrità del database.	Verifica la connettività e l'integrità del database delle istanze di test. Quindi contrassegna i server di origine come Ready for cutover nella console AWS MGN.	Amministratore cloud, ingegnere addetto alla migrazione

Attività preliminari alla migrazione

Operazione	Description	Competenze richieste
Verifica l'integrità del database.	Questo aiuta a garantire che non vi siano problemi di integrità del database all'origine prima della migrazione. Esegui `DBCC CHECKDB` e specifica `WITH_PHYSICAL_ONLY`. L'esecuzione di questo controllo senza autorizzazione `WITH_PHYSICAL_ONLY` può causare problemi di prestazioni all'origine. Per mantenere l'integrità del database, esegui un controllo completo settimanale del database. Questi comandi controllano l'integrità logica e fisica del database rilevando potenziali problemi di danneggiamento.	Ingegnere dei dati, DBA

Operazione	Description	Competenze richieste
	Il controllo verifica la struttura del database, ad esempio pagine, righe, indici e tabelle di sistema.	
Verifica le connessioni ai server collegati.	Verifica le connessioni tra tutti i server esistenti e documenta il loro stato. Questo aiuta a garantire che i server collegati funzionino come previsto dopo la migrazione.	Ingegnere dei dati, DBA
Verifica i backup.	Conferma l'integrità dei backup di origine.	Ingegnere dei dati, DBA

AWS Cutover MGN

Operazione	Description	Competenze richieste
Arresta SQL Server e i servizi del cluster.	Arresta i servizi cluster SQL Server e Microsoft su tutti i nodi del cluster SQL.	DBA, ingegnere addetto alla migrazione
Verifica i server.	Nella console AWS MGN, verificate che lo stato dei server di origine sia Pronto per il cutover e che lo stato della replica dei dati sia Integrato.	Ingegnere della migrazione
Avvia il cutover.	1. Avviate il cutover AWS MGN per il server di controllo. 2. Avvia il cutover AWS MGN per le istanze del cluster SQL Always On.	Ingegnere della migrazione

Operazione	Description	Competenze richieste
	3. Verifica che lo stato passi a Cutover in corso. Per ulteriori informazioni, vedete Avvio di un'istanza cutover nella documentazione di MGN. AWS	
Prova i server lanciati.	Accedi alle EC2 istanze Amazon avviate e convalida lo stato del cluster. Verifica che i server si trovino nelle sottoreti corrette, che le dimension i dell'istanza e le impostazi oni IOPS siano corrette e che il server di controllo sia accessibile.	DBA, ingegnere addetto alla migrazione

Attività successive all'installazione del database

Operazione	Description	Competenze richieste
Aggiorna l'indirizzo IP del cluster.	Aggiorna l'indirizzo IP del cluster per il cluster Windows utilizzando i due indirizzi IP riservati nelle sottoreti di destinazione. Per ulteriori informazioni, vedere Modifica dell'indirizzo IP di un'istanza del cluster di failover.	DBA, ingegnere addetto alla migrazione
Aggiorna il listener IPs del gruppo di disponibilità Always On.	1. Aprire il gestore del cluster di failover.	DBA, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	2. Scegli il ruolo del gruppo di disponibilità Always On. 3. Espandi il nome del listener del gruppo di disponibilità. 4. Nel menu contestuale (clic con il pulsante destro del mouse), scegliete Proprietà dell'indirizzo IP. 5. Aggiorna l'indirizzo IP utilizzando l'indirizzo riservato al listener nella sottorete di destinazione. 6. Connect all'istanza primaria di SQL Server tramite SSMS e verifica che il listener Always On utilizzi entrambe le sottoreti. IPs	
Verifica la connessione.	Utilizzando SSMS, connettiti al listener del gruppo di disponibilità Always On e conferma che la connessione abbia esito positivo.	DBA, ingegnere addetto alla migrazione
Verifica lo stato del gruppo di disponibilità Always On.	1. Vai alla cartella dei gruppi di disponibilità, apri il menu contestuale (fai clic con il pulsante destro del mouse) e scegli Mostra dashboard. 2. Per tutte le repliche, verifica che lo stato di sincroniz zazione sia Sincronizzato.	DBA, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
Controlla il registro degli errori.	Aprire il log degli errori e verificare gli eventuali errori segnalati per l'istanza di SQL Server. Assicurarsi che il ripristino sia completo per tutti i database.	DBA, ingegnere addetto alla migrazione
Prova i server collegati.	Verifica la connettività di tutti i server collegati. In caso di problemi di connettività, assicurati che il server e la porta di destinazione siano accessibili.	DBA, ingegnere addetto alla migrazione

Finalizza il cutover

Operazione	Description	Competenze richieste
Finalizza il cutover.	Dopo aver convalidato i cluster SQL Always On di destinazione, finalizza il cutover tramite la console MGN. AWS In questo modo si interrompe la replica dei dati dai server di origine e si eliminano i dati dal server di replica. Rimuove inoltre il server di replica e le risorse associate.	Amministratore del cloud, ingegnere addetto alla migrazione

risoluzione dei problemi

Problema	Soluzione
AWS Risoluzione dei problemi MGN	Per problemi e risoluzioni comuni, consultate le sezioni Risoluzione dei problemi e Domande frequenti nella documentazione di AWS MGN.

Risorse correlate

AWS resources

- Opzione 1 Rehost - () AWS Application Migration ServiceAWS
- Che cos'è? AWS Application Migration Service

Risorse SQL Server

- Che cos'è SQL Server Management Studio (SSMS)?

Informazioni aggiuntive

Per i requisiti di sicurezza standard per la migrazione dei carichi di lavoro a Cloud AWS, consulta le migliori pratiche per la sicurezza, l'identità e la conformità sul sito Web. AWS

Esegui la migrazione di SQL Server su AWS utilizzando gruppi di disponibilità distribuiti

Praveen Marthala, Amazon Web Services

Riepilogo

I gruppi di disponibilità Always On di Microsoft SQL Server forniscono una soluzione ad alta disponibilità (HA) e disaster recovery (DR) per SQL Server. Un gruppo di disponibilità è costituito da una replica primaria che accetta read/write traffico e fino a otto repliche secondarie che accettano traffico di lettura. Un gruppo di disponibilità è configurato su un Windows Server Failover Cluster (WSFC) con due o più nodi.

I gruppi di disponibilità distribuita Microsoft SQL Server Always On forniscono una soluzione per configurare due gruppi di disponibilità separati tra due indipendenti WFSCs. I gruppi di disponibilità che fanno parte del gruppo di disponibilità distribuita non devono necessariamente trovarsi nello stesso data center. Un gruppo di disponibilità può essere locale e l'altro gruppo di disponibilità può trovarsi su istanze Amazon Web Services (AWS) Cloud su Amazon Elastic Compute Cloud (Amazon EC2) in un dominio diverso.

Questo modello descrive i passaggi per l'utilizzo di un gruppo di disponibilità distribuito per migrare i database SQL Server locali che fanno parte di un gruppo di disponibilità esistente a SQL Server con gruppi di disponibilità configurati su Amazon. EC2 Seguendo questo schema, puoi migrare i database sul cloud AWS con tempi di inattività minimi durante il cutover. I database sono altamente disponibili su AWS subito dopo il cutover. Puoi anche utilizzare questo modello per modificare il sistema operativo sottostante da locale ad AWS mantenendo la stessa versione di SQL Server.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- AWS Direct Connect o AWS Site-to-Site VPN
- La stessa versione di SQL Server installata in locale e sui due nodi di AWS

Versioni del prodotto

- SQL Server versione 2016 e successive

- SQL Server Enterprise Edition

Architecture

Stack tecnologico di origine

- Database Microsoft SQL Server con gruppi di disponibilità Always On locali

Stack tecnologico Target

- Database Microsoft SQL Server con gruppi di disponibilità Always On EC2 su Amazon sul cloud AWS

Architettura di migrazione

Terminologia

- WSFC 1: WSFC locale
- WSFC 2 — WSFC sul cloud AWS
- AG 1 — Primo gruppo di disponibilità, che si trova in WSFC 1
- AG 2 — Secondo gruppo di disponibilità, che si trova nel WSFC 2
- Replica primaria di SQL Server: nodo in AG 1 considerato il principale globale per tutte le scritture
- SQL Server forwarder: nodo in AG 2 che riceve i dati in modo asincrono dalla replica primaria di SQL Server
- Replica secondaria di SQL Server: nodi in AG 1 o AG 2 che ricevono dati in modo sincrono dalla replica primaria o dal server d'inoltro

Tools (Strumenti)

- AWS Direct Connect: AWS Direct Connect collega la rete interna a una posizione AWS Direct Connect tramite un cavo Ethernet standard in fibra ottica. Con questa connessione, puoi creare interfacce virtuali direttamente ai servizi AWS pubblici, aggirando i provider di servizi Internet nel tuo percorso di rete.

- Amazon EC2 — Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi usare Amazon EC2 per lanciare tutti o pochi server virtuali di cui hai bisogno, con scalabilità orizzontale o orizzontale.
- AWS Site-to-Site VPN: AWS Site-to-Site VPN supporta la creazione di una rete privata site-to-site virtuale (VPN). Puoi configurare la VPN per far passare il traffico tra le istanze che avvii su AWS e la tua rete remota.
- Microsoft SQL Server Management Studio — Microsoft SQL Server Management Studio (SSMS) è un ambiente integrato per la gestione dell'infrastruttura SQL Server. Fornisce un'interfaccia utente e un gruppo di strumenti con editor di script avanzati che interagiscono con SQL Server.

Epiche

Configura un secondo gruppo di disponibilità su AWS

Operazione	Description	Competenze richieste
Crea un WSFC su AWS.	Crea WSFC 2 su EC2 istanze Amazon con due nodi per HA. Utilizzerai questo cluster di failover per creare il secondo gruppo di disponibilità (AG 2) su AWS.	Amministratore di sistema, SysOps amministratore
Creare il secondo gruppo di disponibilità su WSFC 2.	Utilizzando SSMS, crea AG 2 su due nodi in WSFC 2. Il primo nodo in WSFC 2 fungerà da server d'inoltro. Il secondo nodo di WSFC 2 fungerà da replica secondaria di AG 2. In questa fase, nessun database è disponibile in AG 2. Questo è il punto di partenza per configurare il	DBA, Sviluppatore

Operazione	Description	Competenze richieste
	gruppo di disponibilità distribui to.	
Crea database senza opzioni di ripristino su AG 2.	Esegui il backup dei database nel gruppo di disponibilità locale (AG 1). Ripristina i database sia sul server d'inoltro che sulla replica secondaria di AG 2 senza alcuna opzione di ripristino. Durante il ripristino dei database, specificate una posizione con spazio su disco sufficiente per i file di dati del database e i file di registro. In questa fase, i database sono in fase di ripristino. Non fanno parte di AG 2 o del gruppo di disponibilità distribui ta e non si sincronizzano.	DBA, Sviluppatore

Configurare il gruppo di disponibilità distribuito

Operazione	Description	Competenze richieste
Crea il gruppo di disponibilità distribuita su AG 1.	Per creare il gruppo di disponibilità distribuita su AG 1, utilizzare l'`DISTRIBUT ED` opzione `CREATE AVAILABILITY GROUP` with.	DBA, Sviluppatore

Operazione	Description	Competenze richieste
	1. Utilizza gli indirizzi degli `LISTENER_URL` endpoint per AG 1 e AG 2. 2. `AVAILABILITY-MODE`Utilizzatelo infatti `ASYNCHRONOUS_COMMIT` per evitare l'eventuale latenza di rete. Ciò non influirà sulle prestazioni del database. 3. Per `FAILOVER_MODE`, utilizza `MANUAL`. È l'unica modalità di disponibilità che funziona con i gruppi di disponibilità distribuiti. 4. Per ripristinare manualmente i database su AG 2 e avere un maggiore controllo su database più grandi, usa `MANUAL` for`SEEDING_MODE`.	

Operazione	Description	Competenze richieste
Crea il gruppo di disponibilità distribuita su AG 2.	Per creare il gruppo di disponibilità distribuita su AG 2, utilizzare `ALTER AVAILABILITY GROUP` con l'`DISTRIBUTED` opzione. 1. Utilizza gli indirizzi degli `LISTENER_URL` endpoint per AG 1 e AG 2. 2. `AVAILABILITY-MODE`Utilizzatelo infatti `ASYNCHRONOUS_COMMIT` per evitare l'eventuale latenza di rete. Ciò non influirà sulle prestazioni del database. 3. Per `FAILOVER_MODE` , utilizza `MANUAL`. È l'unica modalità di disponibilità che funziona con i gruppi di disponibilità distribuiti. 4. Per ripristinare manualmente i database su AG 2 e avere un maggiore controllo su database più grandi, usa `MANUAL` for`SEEDING_MODE` . Il gruppo di disponibilità distribuito viene creato tra AG 1 e AG 2. I database di AG 2 non sono ancora configurati per	DBA, Sviluppatore

Operazione	Description	Competenze richieste
	partecipare al flusso di dati da AG 1 a AG 2.	
Aggiungi database al server d'inoltro e alla replica secondaria su AG 2.	Aggiungi i database al gruppo di disponibilità distribuito utilizzando l'`ALTER DATABASESET HADRAVAILABILITY GROUP`opzione sia nel server d'inoltro che nella replica secondaria su AG 2. Ciò avvia il flusso di dati asincrono tra i database su AG 1 e AG 2. Il primario globale esegue le scritture, invia i dati in modo sincrono alla replica secondari a su AG 1 e invia i dati in modo asincrono al server d'inoltro su AG 2. Lo spedizion iere su AG 2 invia i dati in modo sincrono alla replica secondaria su AG 2.	DBA, Sviluppatore

Monitora il flusso di dati asincrono tra AG 1 e AG 2

Operazione	Description	Competenze richieste
Utilizzo DMVs e registri di SQL Server.	Monitora lo stato del flusso di dati tra due gruppi di disponibilità utilizzando viste di gestione dinamiche (DMVs) e log di SQL Server.	DBA, Sviluppatore

Operazione	Description	Competenze richieste
	DMVs che sono interessanti per il monitoraggio includono `sys.dm_hadr_availability_replica_states` e `sys.dm_hadr_automatic_seeding`. Per lo stato della sincronizzazione del server d'inoltro, monitora lo stato sincronizzato nel registro di SQL Server sul server d'inoltro.	

Esegui attività complementari per la migrazione finale

Operazione	Description	Competenze richieste
Interrompi tutto il traffico verso la replica principale.	Blocca il traffico in entrata verso la replica principale in AG 1 in modo che non si verifichi alcuna attività di scrittura sui database e che i database siano pronti per la migrazione.	Proprietario dell'app, sviluppatore
Modifica la modalità di disponibilità del gruppo di disponibilità distribuita su AG 1.	Nella replica principale, imposta la modalità di disponibilità del gruppo di disponibilità distribuita su sincrona. Dopo aver modificato la modalità di disponibilità in sincrona, i dati vengono inviati	DBA, Sviluppatore

Operazione	Description	Competenze richieste
	in modo sincrono dalla replica primaria in AG 1 al server d'inoltro in AG 2.	
Controlla LSNs in entrambi i gruppi di disponibilità.	Controlla gli ultimi numeri di sequenza di log (LSNs) sia in AG 1 che in AG 2. Poiché non viene eseguita alcuna scrittura nella replica primaria in AG 1, i dati sono sincronizzati e la durata LSNs per entrambi i gruppi di disponibilità deve corrispondere.	DBA, Sviluppatore
Aggiorna AG 1 al ruolo secondario.	Quando aggiorni AG 1 al ruolo secondario, AG 1 perde il ruolo di replica principale e non accetta scritture e il flusso di dati tra due gruppi di disponibilità si interrompe.	DBA, Sviluppatore

Failover verso il secondo gruppo di disponibilità

Operazione	Description	Competenze richieste
Failover manuale su AG 2.	Sul forwarder di AG 2, modificate il gruppo di disponibilità distribuita per consentire la perdita dei dati. Poiché avete già verificato e confermato che gli ultimi dati LSNs su AG 1 e AG 2 coincidono, la perdita di dati non è un problema.	DBA, Sviluppatore

Operazione	Description	Competenze richieste
	Quando si consente la perdita di dati sullo spedizioniere in AG 2, i ruoli di AG 1 e AG 2 cambiano: • AG 2 diventa il gruppo di disponibilità con la replica principale e la replica secondaria. • AG 1 diventa il gruppo di disponibilità con lo spedizion iere e la replica secondaria.	
Modifica la modalità di disponibilità del gruppo di disponibilità distribuita su AG 2.	Sulla replica principale in AG 2, modifica la modalità di disponibilità in asincrona. Ciò modifica lo spostamento dei dati da AG 2 a AG 1, da sincrono a asincrono. Questo passaggio è necessario per evitare l'eventuale latenza di rete tra AG 2 e AG 1 e non influirà sulle prestazioni del database.	DBA, Sviluppatore

Operazione	Description	Competenze richieste
Inizia a inviare traffico alla nuova replica primaria.	Aggiorna la stringa di connessione per utilizzare l'endpoint URL del listener su AG 2 per inviare traffico ai database. AG 2 ora accetta scritture e invia dati allo spedizioniere in AG 1, oltre all'invio di dati alla propria replica secondaria in AG 2. I dati vengono trasferiti in modo asincrono da AG 2 a AG 1.	Proprietario dell'app, sviluppatore

Esegui attività post-cutover

Operazione	Description	Competenze richieste
Elimina il gruppo di disponibilità distribuita su AG 2.	Monitora la migrazione per il periodo di tempo pianificato. Quindi rilascia il gruppo di disponibilità distribuita su AG 2 per rimuovere la configurazione del gruppo di disponibilità distribuita tra AG 2 e AG 1. Ciò rimuove la configurazione del gruppo di disponibilità distribuito e il flusso di dati da AG 2 a AG 1 si interrompe. A questo punto, AG 2 è altamente disponibile su AWS, con una replica primaria che esegue scritture e una	DBA, Sviluppatore

Operazione	Description	Competenze richieste
	replica secondaria nello stesso gruppo di disponibilità.	
Disattiva i server locali.	Disattiva i server locali in WSFC 1 che fanno parte di AG 1.	Amministratore di sistema, amministratore SysOps

Risorse correlate

- Gruppi di disponibilità distribuiti
- SQL Docs: gruppi di disponibilità distribuiti
- SQL Docs: Always On Availability Groups: una soluzione ad alta disponibilità e disaster recovery

Migrare un database relazionale su MongoDB Atlas su AWS

Battulga Purevragchaa e Igor Alekseev, Amazon Web Services

Babu Srinivasan, MongoDB

Riepilogo

Questo modello descrive i passaggi per la migrazione da un database relazionale come SQL Server, MySQL o PostgreSQL a MongoDB Atlas in. Cloud AWS Utilizza MongoDB Relational Migrator per accelerare la migrazione dei dati dai database relazionali a MongoDB Atlas.

Il modello accompagna la guida Migrating to MongoDB Atlas sul AWS sito web Prescriptive Guidance. AWS Fornisce le fasi di implementazione per uno degli scenari di migrazione descritti in tale guida. Per ulteriori scenari di migrazione, consulta i seguenti modelli sul sito Web AWS Prescriptive Guidance:

- Migrare un ambiente MongoDB ospitato autonomamente su MongoDB Atlas su AWS
- Trasmetti dati da IBM Db2, SAP, Sybase e altri database a MongoDB Atlas su AWS

Il modello è destinato ai partner e agli utenti di AWS System Integrator (SI). AWS

Prerequisiti e limitazioni

Prerequisiti

- Un database relazionale di origine (Oracle Database, SQL Server, PostgreSQL, MySQL, SAP/Sybase ASE e altri) per la migrazione a MongoDB Atlas.
- Familiarità con database relazionali, MongoDB Atlas e. Servizi AWS Questo modello spiega alcune delle fasi di migrazione ad alto livello. Ulteriori dettagli verranno aggiunti nelle versioni future.

Versioni del prodotto

- MongoDB versione 5.0 o successiva

Architecture

Il diagramma seguente mostra la migrazione da un database RDBMS (Relational Database Management System) a MongoDB Atlas on. AWS

Per le architetture di riferimento di MongoDB Atlas che supportano diversi scenari di utilizzo, consulta Migrating to MongoDB Atlas sul sito Web Prescriptive Guidance. AWS AWS

Tools (Strumenti)

- MongoDB Atlas è un database come servizio DBaa (S) completamente gestito per l'implementazione e la gestione dei database MongoDB nel cloud.
- MongoDB Relational Migrator offre una transizione fluida dei dati dai database relazionali tradizionali a MongoDB. Aiuta ad automatizzare il processo di conversione e trasforma il modello di dati strutturati dei database relazionali nel formato di documento flessibile fornito da MongoDB. Relational Migrator preserva l'integrità e le relazioni dei dati per semplificare la migrazione. Le organizzazioni possono sfruttare i vantaggi in termini di scalabilità, prestazioni e versatilità offerti da MongoDB, pur mantenendo la familiarità dei dati esistenti.

Best practice

Per le migliori pratiche per l'utilizzo di MongoDB AWS su, consulta i post sul AWS blog di Partner Network.

Epiche

Scoperta e valutazione

Operazione	Description	Competenze richieste
Determinare i parametri e le dimensioni del database relazionale.	Stima la dimensione del set di lavoro utilizzando i consigli e le informazioni di Relational Migrator `db.stats()` per lo spazio totale dell'indice. Si supponga che si acceda frequentemente a una percentuale dello spazio dati. Questa operazione dovrebbe richiedere circa una settimana. Per ulteriori informazioni	Proprietario dell'app, DBA

Operazione	Description	Competenze richieste
	ed esempi su questa e le altre storie di questa epopea, consulta la sezione Risorse correlate.	
Stima dei requisiti di larghezza di banda della rete.	Per stimare i requisiti di larghezza di banda di rete, moltiplica la dimensione media dei documenti per il numero di documenti inviati al secondo. Considerate come base il traffico massimo che qualsiasi nodo del cluster sarà in grado di sopportare. Per calcolare le velocità di trasferimento dei dati a valle dal cluster alle applicazioni client, utilizza la somma del totale dei documenti restituiti in un periodo di tempo. Se le tue applicazioni leggono da nodi secondari, dividi questo numero di documenti totali per il numero di nodi che possono eseguire operazioni di lettura. Per trovare la dimensione media dei documenti per un database, utilizzate il `db.stats().avgObjSize` comando. Questa operazione richiede in genere un giorno.	DBA

Operazione	Description	Competenze richieste
Seleziona il livello Atlas.	Segui le istruzioni nella documentazione di MongoDB per selezionare il livello di cluster Atlas corretto.	DBA
Piano di taglio.	Piano per il completamento dell'applicazione.	DBA, proprietario dell'app

Configura un nuovo ambiente MongoDB Atlas su AWS

Operazione	Description	Competenze richieste
Crea un nuovo cluster MongoDB Atlas su. AWS	In MongoDB Atlas, scegli Crea un cluster. Nella finestra di dialogo Crea nuovo cluster, seleziona AWS come fornitore di servizi cloud.	DBA
Configurazione Regioni AWS selettiva e globale del cluster.	Seleziona dall'elenco delle opzioni disponibili Regioni AWS per il tuo cluster Atlas. Se necessario, configura i cluster globali.	DBA
Seleziona il livello del cluster.	Seleziona il livello di cluster preferito. La selezione del livello determina fattori come la memoria, lo storage e le specifiche IOPS.	DBA
Configura impostazioni aggiuntive del cluster.	Configura impostazioni del cluster aggiuntive come la versione di MongoDB, il backup e le opzioni di crittogra fia. Per ulteriori informazioni	DBA

Operazione	Description	Competenze richieste
	su queste opzioni, consulta la sezione Risorse correlate.	

Configura sicurezza e conformità

Operazione	Description	Competenze richieste
Configura l'elenco di accesso.	Per connettersi al cluster Atlas, è necessario aggiunger e una voce all'elenco di accesso del progetto. Atlas lo utilizza TLS/SSL per crittogra fare le connessioni al cloud privato virtuale (VPC) per il database. Per impostare la lista di accesso al progetto e per ulteriori informazioni sulle storie di questa epopea, consulta la sezione Risorse correlate.	DBA
Autentica e autorizza gli utenti.	È necessario creare e autenticare gli utenti del database che accederanno ai cluster MongoDB Atlas. Per accedere ai cluster di un progetto, gli utenti devono appartenere a quel progetto e possono appartenere a più progetti.	DBA
Crea ruoli personalizzati.	(Facoltativo) Atlas supporta la creazione di ruoli personali zzati nei casi in cui le autorizzazioni utente integrate	DBA

Operazione	Description	Competenze richieste
	nel database Atlas non coprono il set di autorizzazioni desiderato.	
Configura il peering VPC.	(Facoltativo) Atlas supporta il peering VPC con altri dispositivi. VPCs AWS	Amministratore AWS
Configura un AWS PrivateLink endpoint.	(Facoltativo) Puoi configurare endpoint privati utilizzando AWS . AWS PrivateLink Per ulteriori informazioni, consulta la documentazione di Amazon VPC.	Amministratore AWS
Abilita l'autenticazione a due fattori.	(Facoltativo) Atlas supporta l'autenticazione a due fattori (2FA) per aiutare gli utenti a controllare l'accesso ai propri account Atlas.	Amministratore AWS
Configura l'autenticazione e l'autorizzazione degli utenti con LDAP.	(Facoltativo) Atlas supporta l'esecuzione dell'autenticazione e dell'autorizzazione degli utenti con Lightweight Directory Access Protocol (LDAP).	DBA
Configura l'accesso unificato AWS .	(Facoltativo) Alcune funzionalità di Atlas, tra cui Atlas Data Lake e la crittografia a riposo mediante la gestione delle chiavi dei clienti, utilizzano ruoli AWS Identity and Access Management (IAM) per l'autenticazione.	Amministratore AWS

Operazione	Description	Competenze richieste
Configura la crittografia a riposo utilizzando AWS KMS.	(Facoltativo) Atlas supporta l'utilizzo di AWS Key Management Service (AWS KMS) per crittografare i motori di archiviazione e i backup dei provider di cloud.	Amministratore AWS
Configura la crittografia a livello di campo lato client.	(Facoltativo) Atlas supporta la crittografia a livello di campo lato client, inclusa la crittografia automatica dei campi.	Amministratore AWS

Migrazione dei dati

Operazione	Description	Competenze richieste
Aggiungi MongoDB Relational Migrator all'elenco di accesso.	Aggiungi Relational Migrator all'elenco di accesso per il tuo database di origine. Questo aiuta a preparare l'ambiente di origine per la connessione al cluster Atlas di destinazione.	DBA
Valuta gli oggetti del database relazionale.	Avvia MongoDB Relational Migrator e connettiti al tuo database relazionale. Inizia la valutazione.	DBA
Accetta i modelli di migrazione o scegli di modificarli in base alle tue esigenze aziendali.	Accetta i modelli di database consigliati da Relational Migrator in base alla valutazione iniziale e ai parametri prestazionali oppure scegli di	DBA

Operazione	Description	Competenze richieste
	modificarli in base ai requisiti aziendali.	
Avvia la replica di destinazione impostata in MongoDB Atlas.	Avvia la replica di destinazione impostata in MongoDB Atlas. In Relational Migrator, scegli Sono pronto per la migrazione.	DBA

Configurare l'integrazione operativa

Operazione	Description	Competenze richieste
Connect al cluster MongoDB Atlas.	Assicurati che la connettività del cluster MongoDB Atlas funzioni come previsto.	Proprietario dell'app
Interagisci con i dati del cluster.	Verifica i dati del cluster.	DBA
Monitora i tuoi cluster.	Verifica che i cluster siano configurati correttamente.	DBA
Eseguire il backup e il ripristino dei dati del cluster.	Pianifica i backup con una cadenza regolare per i dati del cluster.	DBA

Risorse correlate

Tutti i seguenti collegamenti, se non diversamente specificato, portano alle pagine Web della documentazione di MongoDB.

Guida alla migrazione

- Migrazione a MongoDB Atlas AWS on (Guida prescrittiva)AWS

Scoperta e valutazione

- Memoria
- Esempio di dimensionamento con set di dati di esempio Atlas
- Esempio di dimensionamento per applicazioni mobili
- Traffico di rete
- Scalabilità automatica del cluster
- Modello di dimensionamento Atlas

Configurazione della sicurezza e della conformità

- Configurazione delle voci dell'elenco di accesso IP
- Configurazione degli utenti del database
- Configura l'accesso all'interfaccia utente Atlas
- Configura i ruoli personalizzati del database
- Configurazione degli utenti del database
- Configurare una connessione peering di rete
- Scopri di più sugli endpoint privati in Atlas
- Gestisci le tue opzioni di autenticazione a più fattori
- Configura l'autenticazione e l'autorizzazione degli utenti con LDAP
- Atlas Data Lake
- Crittografia a riposo con Customer Key Management
- Metodi per assumere un ruolo (documentazione IAM)
- Crittografia a livello di campo lato client
- Crittografia automatica
- Controlli di sicurezza MongoDB Atlas
- Centro fiducia MongoDB
- Configura le funzionalità di sicurezza per i cluster

Configurazione di un nuovo ambiente MongoDB Atlas su AWS

- Provider di cloud e regioni
- Gestisci i cluster globali
- Seleziona il livello del cluster

- Configura impostazioni aggiuntive
- Inizia con Atlas
- Configura l'accesso all'interfaccia utente Atlas

Migrazione dei dati

- Migrazione o importazione dei dati

Cluster di monitoraggio

- Monitora i tuoi cluster

Integrazione delle operazioni

- Connect a un cluster
- Interagisci con i tuoi dati
- Monitora i tuoi cluster
- Backup, ripristino e archiviazione dei dati

Migrare un ambiente MongoDB ospitato autonomamente su MongoDB Atlas su AWS

Battulga Purevragchaa e Igor Alekseev, Amazon Web Services

Babu Srinivasan, MongoDB

Riepilogo

Questo modello descrive i passaggi per la migrazione da un ambiente MongoDB autogestito (incluso MongoDB Community Server, Enterprise Server, Enterprise Advanced, MLab o qualsiasi cluster MongoDB gestito) a MongoDB Atlas nel. Cloud AWS Utilizza il servizio Atlas Live Migration per accelerare la migrazione dei dati da MongoDB a MongoDB Atlas.

Il modello accompagna la guida Migrating to MongoDB Atlas sul AWS sito web Prescriptive Guidance. AWS Fornisce le fasi di implementazione per uno degli scenari di migrazione descritti in tale guida. Per ulteriori scenari di migrazione, consulta i seguenti modelli sul sito Web AWS Prescriptive Guidance:

- Migrare un database relazionale su MongoDB Atlas su AWS
- Trasmetti dati da IBM Db2, SAP, Sybase e altri database a MongoDB Atlas su AWS

Il modello è destinato ai partner e agli utenti di AWS Systems Integrator (SI). AWS

Prerequisiti e limitazioni

Prerequisiti

- Un MongoDB Enterprise Advanced, Community Server di origine o altro ambiente MongoDB autogestito per la migrazione a MongoDB Atlas.
- Familiarità con MongoDB, MongoDB Atlas e. Servizi AWS Questo modello spiega alcune delle fasi di migrazione ad alto livello. Ulteriori dettagli verranno aggiunti nelle versioni future.

Versioni del prodotto

- MongoDB versione 6.0.13 o successiva

Architecture

Il diagramma seguente mostra l'Atlas Live Migration Service utilizzato per migrare i dati da un database MongoDB Enterprise Advanced e da un database della comunità MongoDB a MongoDB Atlas on. AWS Utilizza questo servizio quando devi migrare database complessi e di grandi dimensioni su MongoDB Atlas con tempi di inattività minimi e sincronizzazione continua dei dati. Questo modello utilizza Atlas Live Migration Service.

Il diagramma seguente mostra il servizio mirror MongoDB `mongomirror` () che è possibile utilizzare anche per migrare i dati da un database MongoDB Enterprise Advanced e da un database della comunità MongoDB a MongoDB Atlas tramite una connessione protetta. AWS AWS PrivateLink Utilizzalo `mongomirror` per la replica continua dei dati tra MongoDB locali e MongoDB Atlas. Questo strumento è ideale per il disaster recovery o le migrazioni a fasi, ma non rientra nell'ambito di questo modello.

Per ulteriori architetture di riferimento di MongoDB Atlas che supportano diversi scenari di utilizzo, consulta Migrazione a MongoDB Atlas sul sito Web Prescriptive Guidance. AWS AWS

Tools (Strumenti)

- MongoDB Atlas è un database as a service (DBaaS) completamente gestito per l'implementazione e la gestione di database MongoDB nel cloud.
- Atlas Live Migration Service è un'utilità MongoDB gratuita che aiuta a migrare i database su Atlas. Questo servizio mantiene il database di origine sincronizzato con il database di destinazione fino al cutover. Quando si è pronti per il cutover, si interrompono le istanze dell'applicazione, le si indirizza al cluster Atlas di destinazione e le si riavvia. Per accedere a questo servizio, scegli le opzioni del database dal tuo cluster MongoDB Atlas.
- mongomirror è uno strumento per la migrazione manuale dei dati da un set di repliche MongoDB esistente a un set di repliche MongoDB Atlas. `mongomirror`non richiede la chiusura del set di repliche o delle applicazioni esistenti, non importa i dati degli utenti o dei ruoli né copia il database di configurazione. È possibile effettuare il download `mongomirror` dalla documentazione di MongoDB.

Best practice

Per le migliori pratiche per l'utilizzo di MongoDB AWS su, consulta i post sul AWS blog di Partner Network.

Epiche

Scoperta e valutazione

Operazione	Description	Competenze richieste
Determina la dimensione del cluster.	Stimate la dimensione del set di lavoro utilizzando le informazioni fornite `db.stats()` per lo spazio totale dell'indice. Si supponga che si acceda frequentemente a una percentuale dello spazio dati. In alternativa, è possibile stimare i requisiti di memoria in base alle proprie ipotesi. Questa operazione dovrebbe richiedere circa una settimana. Per ulteriori informazioni ed esempi su questa e le altre storie di questa epopea, consulta la sezione Risorse correlate.	DBA, proprietario dell'app
Stima dei requisiti di larghezza di banda della rete.	Per stimare i requisiti di larghezza di banda di rete, moltiplica la dimensione media dei documenti per il numero di documenti inviati al secondo. Considerate come base il traffico massimo che qualsiasi nodo del cluster sarà in grado di sopportare.	DBA

Operazione	Description	Competenze richieste
	Per calcolare le velocità di trasferimento dei dati a valle dal cluster alle applicazioni client, utilizza la somma del totale dei documenti restituiti in un periodo di tempo. Se le tue applicazioni leggono da nodi secondari, dividi questo numero di documenti totali per il numero di nodi che possono eseguire operazioni di lettura. Per trovare la dimensione media dei documenti per un database, utilizzate il `db.stats().avgObjSize` comando. Questa operazione richiede in genere un giorno.	
Seleziona il livello Atlas.	Segui le istruzioni nella documentazione di MongoDB per selezionare il livello di cluster Atlas corretto.	DBA
Piano per il cutover.	Piano per il completamento dell'applicazione.	DBA, proprietario dell'app

Configura un nuovo ambiente MongoDB Atlas su AWS

Operazione	Description	Competenze richieste
Crea un nuovo cluster MongoDB Atlas su. AWS	Accedi ad Atlas e apri la pagina Panoramica del tuo progetto. Scegli il pulsante Crea per creare un cluster.	DBA

Operazione	Description	Competenze richieste
	Per ulteriori informazioni, consulta la documentazione di MongoDB.	
Configurazione Regioni AWS selettiva e globale del cluster.	Seleziona dall'elenco delle opzioni disponibili Regioni AWS per il tuo cluster Atlas. Se necessario, configura i cluster globali. Per ulteriori informazioni, consulta la documentazione di MongoDB.	DBA
Seleziona il livello del cluster.	Seleziona il tuo livello di cluster preferito. La selezione del livello determina fattori quali memoria, storage e specifiche IOPS.	DBA
Configura impostazioni aggiuntive del cluster.	Configura impostazioni del cluster aggiuntive come la versione di MongoDB, il backup e le opzioni di crittografia. Per ulteriori informazioni su queste opzioni, consulta la sezione Risorse correlate.	DBA

Configura sicurezza e conformità

Operazione	Description	Competenze richieste
Autentica e autorizza gli utenti.	È necessario creare e autenticare gli utenti del database che accederanno ai cluster MongoDB Atlas. Per accedere ai cluster di un	DBA

Operazione	Description	Competenze richieste
	progetto, gli utenti devono appartenere a quel progetto e possono appartenere a più progetti. Atlas supporta anche l'autenticazione basata su AWS Identity and Access Management (IAM). Per ulteriori informazioni, consulta la documentazione di MongoDB.	
Crea ruoli personalizzati.	(Facoltativo) Atlas supporta la creazione di ruoli personalizzati nei casi in cui le autorizzazioni utente integrate nel database Atlas non coprono il set di autorizzazioni desiderato.	DBA
Configura il peering VPC.	(Facoltativo) Atlas supporta il peering su cloud privato virtuale (VPC) con altri. VPCs AWS	Amministratore AWS
Configura un AWS PrivateLink endpoint.	(Facoltativo) Puoi configurare endpoint privati utilizzando AWS . AWS PrivateLink Per ulteriori informazioni, consulta la documentazione di Amazon VPC.	Amministratore AWS
Abilita l'autenticazione a due fattori.	(Facoltativo) Atlas supporta l'autenticazione a due fattori (2FA) per aiutare gli utenti a controllare l'accesso ai propri account Atlas.	Amministratore AWS

Operazione	Description	Competenze richieste
Configura l'autenticazione e l'autorizzazione degli utenti con LDAP.	(Facoltativo) Atlas supporta l'esecuzione dell'autenticazione e dell'autorizzazione degli utenti con Lightweight Directory Access Protocol (LDAP).	Amministratore AWS
Configura l' AWS accesso unificato.	(Facoltativo) Alcune funzionalità di Atlas, tra cui Atlas Data Lake e la crittografia a riposo mediante la gestione delle chiavi dei clienti, utilizzano i ruoli IAM per l'autenticazione.	Amministratore AWS
Configura la crittografia a riposo utilizzando AWS KMS.	(Facoltativo) Atlas supporta l'utilizzo di AWS Key Management Service (AWS KMS) per crittografare i motori di archiviazione e i backup dei provider di cloud.	Amministratore AWS
Configura la crittografia a livello di campo lato client.	(Facoltativo) Atlas supporta la crittografia a livello di campo lato client, inclusa la crittografia automatica dei campi.	Amministratore AWS

Migrazione dei dati

Operazione	Description	Competenze richieste
Seleziona la replica di destinazione impostata in MongoDB Atlas.	Passa al cluster Atlas di destinazione e scegli il pulsante con i puntini di sospensione (...). Nell'elenco dei cluster, questo pulsante	DBA

Operazione	Description	Competenze richieste
	appare sotto il nome del cluster. Nei dettagli del cluster, il pulsante appare sulla destra, accanto ai pulsanti Connect e Configuration. Per ulteriori informazioni, consulta la documentazione di MongoDB.	
Aggiungi Atlas Live Migration Service all'elenco di accesso.	Aggiungi Atlas Live Migration Service all'elenco di accesso nel cluster AWS di origine. Questo aiuta a preparare l'ambiente di origine per la connessione al cluster Atlas di destinazione.	DBA
Esegui la migrazione con Atlas Live Migration Service.	Scegli Avvia migrazione. Quando il pulsante Prepare to Cutover diventa verde, esegui il cutover. Rivedi le metriche delle prestazioni del cluster Atlas. Prendi in considera zione l'aggiornamento della connessione al database in tutti i livelli dell'applicazione in modo che punti al nuovo database.	DBA

Configurare l'integrazione operativa

Operazione	Description	Competenze richieste
Connect al cluster MongoDB Atlas.	Assicurati che la connettività del cluster MongoDB Atlas funzioni come previsto.	Proprietario dell'app

Operazione	Description	Competenze richieste
Interagisci con i dati del cluster.	Esegui il test dei dati del cluster.	DBA
Monitora i tuoi cluster.	Verifica che i cluster siano configurati correttamente.	DBA
Eseguire il backup e il ripristino dei dati del cluster.	Pianifica i backup con una cadenza regolare per i dati del cluster.	DBA

Risoluzione dei problemi

Problema	Soluzione

Errore: impossibile raggiungere la fonte specificata	• Assicurati di aver aggiunto gli intervalli di sottoreti corretti all'elenco di accesso IP sul cluster di origine. È possibile trovare i quattro intervalli di sottoreti richiesti nella finestra modale di Live Migration.
	• Conferma che il nome host che hai specificato si risolva in un indirizzo IP pubblico. Al prompt dei comandi, utilizzate uno dei seguenti comandi:
	`nslookup <hostname>`
	`ping <hostname>`
	• Assicurati di non utilizzare una connessione peering VPC, che non è compatibile con pull live migration. Se la tua unica opzione è una connessione peering VPC, usala al suo posto. `mongomirror`

Problema	Soluzione
Errore: impossibile risolvere il nome host	Non è stato trovato alcun indirizzo IP per il nome host specificato. Verifica che il nome host fornito sia corretto e accessibile pubblicamente.
Qualsiasi altro errore	Se riscontri altri errori, consulta Risoluzione dei problemi di Live Migration (Pull) nella documentazione di MongoDB.

Risorse correlate

Tutti i seguenti collegamenti, se non diversamente specificato, portano alle pagine Web della documentazione di MongoDB.

Guida alla migrazione

- Migrazione a MongoDB Atlas AWS on (Guida prescrittiva)AWS

Migrazione legacy

- Migrazione di versioni precedenti di MongoDB

Scoperta e valutazione

- Memoria
- Esempio di dimensionamento con set di dati di esempio Atlas
- Esempio di dimensionamento per applicazioni mobili
- Traffico di rete
- Scalabilità automatica del cluster
- Modello di dimensionamento Atlas

Configurazione della sicurezza e della conformità

- Configurazione delle voci dell'elenco di accesso IP
- Configurazione degli utenti del database

- Configura l'accesso all'interfaccia utente Atlas
- Configura i ruoli personalizzati del database
- Configurazione degli utenti del database
- Configurare una connessione peering di rete
- Scopri di più sugli endpoint privati in Atlas
- Gestisci le tue opzioni di autenticazione a più fattori
- Configura l'autenticazione e l'autorizzazione degli utenti con LDAP
- Atlas Data Lake
- Crittografia a riposo con Customer Key Management
- Metodi per assumere un ruolo (documentazione IAM)
- Crittografia a livello di campo lato client
- Crittografia automatica
- Controlli di sicurezza MongoDB Atlas
- MongoDB Trust Center
- Configura le funzionalità di sicurezza per i cluster

Configurazione di un nuovo ambiente MongoDB Atlas su AWS

- Provider di cloud e regioni
- Gestisci i cluster globali
- Seleziona il livello del cluster
- Configura impostazioni aggiuntive
- Inizia con Atlas
- Configura l'accesso all'interfaccia utente Atlas

Migrazione dei dati

- Migrazione o importazione dei dati

Cluster di monitoraggio

- Monitora i tuoi cluster

Integrazione delle operazioni

- Connect a un cluster
- Interagisci con i tuoi dati
- Monitora i tuoi cluster
- Backup, ripristino e archiviazione dei dati

Addestramento

- Migrazione in tempo reale con MongoDB Atlas

Informazioni aggiuntive

Per ulteriori informazioni, consulta i seguenti argomenti nella documentazione di MongoDB:

- Per spostare i dati su un'istanza serverless, usa Compass per esportare e importare dati o migrare i dati con strumenti autogestiti. Per ulteriori informazioni, consulta Limitazioni delle istanze Serverless.
- Per caricare i dati in un nuovo cluster in Atlas, consulta Caricare dati in Atlas.
- Per creare una copia del cluster a scopo di test, consulta Metodi di backup per una distribuzione autogestita.
- Se l'applicazione che desideri migrare richiede un uptime quasi continuo, contatta il supporto di MongoDB e condividi i tuoi requisiti di uptime e la configurazione del cluster.
- Per ulteriori informazioni, consulta Migrare o importare dati.

Esegui la migrazione di un database Oracle ad Amazon DynamoDB utilizzando AWS DMS

Rambabu Karnena, Amazon Web Services

Riepilogo

Questo modello illustra i passaggi per la migrazione di un database Oracle ad Amazon DynamoDB utilizzando AWS Database Migration Service (AWS DMS). Copre tre tipi di database di origine:

- Database Oracle locali
- Database Oracle su Amazon Elastic Compute Cloud (Amazon EC2)
- Amazon Relational Database Service (Amazon RDS) per istanze database Oracle

In questa dimostrazione di concetto, questo modello si concentra sulla migrazione da un'istanza DB Amazon RDS for Oracle.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un'applicazione che si connette a un database Amazon RDS for Oracle
- Una tabella creata nel database Amazon RDS for Oracle di origine con una chiave primaria e dati di esempio

Limitazioni

- Gli oggetti del database Oracle, come procedure, funzioni, pacchetti e trigger, non vengono presi in considerazione per la migrazione perché Amazon DynamoDB non supporta questi oggetti di database.

Versioni del prodotto

- Questo modello si applica a tutte le edizioni e versioni dei database Oracle supportate da AWS DMS. Per ulteriori informazioni, consulta Utilizzo di un database Oracle come origine per AWS DMS e utilizzo di un database Amazon DynamoDB come destinazione per AWS DMS. Ti

consigliamo di utilizzare le versioni più recenti di AWS DMS per il supporto di versioni e funzionalità più completo.

Architecture

Stack tecnologico di origine

- Amazon RDS per istanze DB Oracle, Oracle su EC2 Amazon o database Oracle locali

Stack tecnologico Target

- Amazon DynamoDB

Architettura di migrazione dei dati AWS

Tools (Strumenti)

- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS. Questo modello utilizza Amazon RDS for Oracle.

Epiche

Pianifica la migrazione

Operazione	Description	Competenze richieste
Crea un VPC.	Nel tuo account AWS, crea un cloud privato virtuale (VPC) e una sottorete privata.	Amministratore di sistema

Operazione	Description	Competenze richieste
Crea gruppi di sicurezza ed elenchi di controllo degli accessi alla rete.	Per ulteriori informazioni, consulta la documentazione di AWS.	Amministratore di sistema
Configura e avvia l'istanza DB di Amazon RDS for Oracle.	Per ulteriori informazioni, consulta la documentazione di AWS.	DBA, amministratore di sistema

Migrazione dei dati

Operazione	Description	Competenze richieste
Crea un ruolo IAM per accedere a DynamoDB.	Nella console AWS Identity and Access Management (IAM), crea il ruolo, allega la policy `AmazonDynamoDBFullAccess to it` e seleziona AWS DMS come servizio.	Amministratore di sistema
Crea un'istanza di replica AWS DMS per la migrazione.	L'istanza di replica deve trovarsi nella stessa zona di disponibilità e nello stesso VPC del database di origine.	Amministratore di sistema
Crea endpoint di origine e destinazione in AWS DMS.	Per creare l'endpoint del database di origine, hai due opzioni: • Sulla console Amazon RDS, scegli Databases, DB identifier, Connectivity & Security e scegli l'endpoint.	Amministratore di sistema

Operazione	Description	Competenze richieste
	• Sulla console AWS DMS, scegli Select RDS DB instance. Per creare l'endpoint del database di destinazione, scegli il ruolo Amazon Resource Name (ARN) dal task precedente per accedere a DynamoDB.	
Crea un task AWS DMS per caricare le tabelle del database Oracle di origine su DynamoDB.	Scegli i nomi degli endpoint di origine e destinazione e l'istanza di replica dai passaggi precedenti. Il tipo può essere a pieno carico. Scegli lo schema Oracle e specifica% per selezionare tutte le tabelle.	Amministratore di sistema
Convalida le tabelle in DynamoDB.	Per visualizzare i risultati della migrazione, scegli Tabelle dal riquadro di navigazio ne a sinistra nella console DynamoDB.	DBA

Migrare l'applicazione

Operazione	Description	Competenze richieste
Modifica il codice dell'appl icazione.	Per connetterti e recuperare dati da DynamoDB, aggiorna il codice dell'applicazione.	Proprietario dell'app, DBA, amministratore di sistema

Tagliare

Operazione	Description	Competenze richieste
Cambia i client dell'appl icazione per utilizzare DynamoDB.		DBA, proprietario dell'app, amministratore di sistema

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse AWS.	Ad esempio, chiudono l'istanza Amazon RDS for Oracle, DynamoDB e l'istanza di replica AWS DMS.	DBA, amministratore di sistema
Raccogli le metriche.	Le metriche includono il tempo necessario per la migrazion e, le percentuali del lavoro manuale e del lavoro svolto dallo strumento e i risparmi sui costi.	DBA, proprietario dell'app, amministratore di sistema

Risorse correlate

- AWS Database Migration Service e Amazon DynamoDB: cosa devi sapere (post sul blog)
- Utilizzo di un database Oracle come sorgente per AWS DMS
- Utilizzo di un database Amazon DynamoDB come destinazione per AWS Database Migration Service
- Best practice per la migrazione da RDBMS ad Amazon DynamoDB (white paper)

Esegui la migrazione da Oracle 8i o 9i ad Amazon RDS for Oracle utilizzando AWS DMS SharePlex

Ramu Jagini, Amazon Web Services

Riepilogo

Questo modello descrive come migrare un database Oracle 8i o 9i locale a un database Amazon Relational Database Service (Amazon RDS) per Oracle. Puoi utilizzare questo modello per completare la migrazione con tempi di inattività ridotti utilizzando Quest per la replica sincrona. SharePlex

È necessario utilizzare un'istanza di database Oracle intermedia per la migrazione perché AWS Database Migration Service (AWS DMS) non supporta Oracle 8i o 9i come ambiente di origine. Puoi utilizzare la versione SharePlex 7.6.3 per eseguire la replica da versioni precedenti del database Oracle a versioni successive del database Oracle. L'istanza intermedia del database Oracle è compatibile come destinazione per la versione SharePlex 7.6.3 e supportata come origine per AWS DMS o versioni più recenti di. SharePlex Questo supporto consente la replica successiva dei dati nell'ambiente di destinazione Amazon RDS for Oracle.

Tieni presente che diversi tipi di dati e funzionalità obsoleti possono influire sulla migrazione da Oracle 8i o 9i alla versione più recente di Oracle Database. Per mitigare questo impatto, questo modello utilizza Oracle 11.2.0.4 come versione intermedia del database per aiutare a ottimizzare il codice dello schema prima della migrazione all'ambiente di destinazione Amazon RDS for Oracle.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database Oracle 8i o 9i di origine in un ambiente locale
- Oracle Database 12c Release 2 (12CR2) per lo staging su Amazon Elastic Compute Cloud (Amazon) EC2
- Quest SharePlex 7.6.3 (livello commerciale)

Limitazioni

- Limitazioni di RDS per Oracle

Versioni del prodotto

- Oracle 8i o 9i per il database di origine
- Oracle 12 CR2 per il database dell'area di gestione temporanea (deve corrispondere alla versione di Amazon RDS for Oracle)
- Oracle 12 CR2 o versione successiva per il database di destinazione (Amazon RDS for Oracle)

Architecture

Stack tecnologico di origine

- Database Oracle 8i o 9i
- SharePlex

Stack tecnologico Target

- Amazon RDS per Oracle

Architettura di migrazione

Il diagramma seguente mostra come migrare un database Oracle 8i o 9i da un ambiente locale a un'istanza DB Amazon RDS for Oracle nel cloud AWS.

Il diagramma mostra il flusso di lavoro seguente:

1. Abilita il database di origine Oracle con la modalità di registro degli archivi, la registrazione forzata e la registrazione supplementare.
2. Ripristina il database di gestione temporanea Oracle dal database di origine Oracle utilizzando il ripristino di Recovery Manager (RMAN) e FLASHBACK_SCN. point-in-time
3. SharePlex Configurare per leggere i redo log dal database di origine Oracle utilizzando (utilizzato in RMAN). `FLASHBACK_SCN`
4. Avvia SharePlex la replica per sincronizzare i dati dal database di origine Oracle al database di gestione temporanea Oracle.
5. Ripristina il database di destinazione Amazon RDS for Oracle utilizzando EXPDP e IMPDP con. `FLASHBACK_SCN`

6. Configura AWS DMS e le relative attività di origine come database di staging Oracle e Amazon RDS for Oracle come database di destinazione utilizzando `FLASHBACK_SCN` (utilizzato in EXPDP).
7. Avvia le attività di AWS DMS per sincronizzare i dati dal database di staging Oracle al database di destinazione Oracle.

Tools (Strumenti)

- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS.
- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali.
- Quest SharePlex è uno strumento di replica Oracle-to-Oracle dei dati per spostare i dati con tempi di inattività minimi e nessuna perdita di dati.
- Recovery Manager (RMAN) è un client di database Oracle che esegue attività di backup e ripristino sui database. Semplifica enormemente il backup, il ripristino e il ripristino dei file di database.
- Data Pump Export consente di caricare dati e metadati in un set di file del sistema operativo chiamato set di file di dump. Il set di file di dump può essere importato solo dall'utilità Data Pump Import o dal pacchetto DBMS_DATAPUMP.

Epiche

Configurazione SharePlex e database di gestione temporanea di Oracle su Amazon EC2

Operazione	Description	Competenze richieste
Crea un' EC2 istanza.	1. Crea un' EC2 istanza. 2. Installa Oracle 12 CR2 sull' EC2 istanza per fungere da database di gestione temporanea di Oracle.	Amministrazione Oracle
Preparare il database dell'area di gestione temporanea.	Prepara il database di gestione temporanea di Oracle per il ripristino come aggiornamento su Oracle 12 CR2 eseguendo il backup	Amministrazione Oracle

Operazione	Description	Competenze richieste
	RMAN dall'ambiente di origine del database Oracle 8i o 9i. Per ulteriori informazioni, vedere Oracle 9i Recovery Manager User's Guide e Database Backup and Recovery User's Guide nella documentazione Oracle.	
Configura SharePlex.	Configura l' SharePlex origine come database Oracle 8i o 9i locale e configura la destinazione come database di CR2 staging Oracle 12 ospitato su Amazon. EC2	SharePlex, Amministrazione Oracle

Configura Amazon RDS for Oracle come ambiente di destinazione

Operazione	Description	Competenze richieste
Crea un'istanza Oracle DB.	Crea un database Amazon RDS for Oracle, quindi collega Oracle CR2 12 al database. Per ulteriori informazioni, consulta Creazione di un'istanza DB Oracle e connessione a un database su un'istanza DB Oracle nella documentazione di Amazon RDS.	DBA
Ripristina Amazon RDS for Oracle dal database di staging.	1. Esegui un backup EXPDP dal server del database di	DBA

Operazione	Description	Competenze richieste
	staging Oracle utilizzando. `FLASHBACK_SCN` 2. Ripristina Amazon RDS for Oracle dal database di staging. Per ulteriori informazioni, consulta 54 DBMS_DATA PUMP nella documentazione Oracle.	

Configura AWS DMS

Operazione	Description	Competenze richieste
Crea endpoint per i database.	Crea un endpoint di origine per il database di staging Oracle e un endpoint di destinazione per il database Amazon RDS for Oracle. Per ulteriori informazioni, consulta Come posso creare endpoint di origine o di destinazione utilizzando AWS DMS? nell'AWS Knowledge Center.	DBA
Crea un'istanza di replica.	Utilizza AWS DMS per avviare un'istanza di replica per il database di staging Oracle sul database Amazon RDS for Oracle.	DBA

Operazione	Description	Competenze richieste
	Per ulteriori informazioni, consulta Come posso creare un'istanza di replica AWS DMS? nell'AWS Knowledge Center.	
Crea e avvia attività di replica.	Crea attività di replica AWS DMS per l'acquisizione dei dati di modifica (CDC) utilizzando `FLASHBACK_SCN` EXPDP (poiché il caricamento completo è già avvenuto tramite EXPDP). Per ulteriori informazioni, consulta Creazione di un'attività nella documentazione di AWS DMS.	DBA

Passare ad Amazon RDS for Oracle

Operazione	Description	Competenze richieste
Interrompi il carico di lavoro dell'applicazione.	Arresta i server delle applicazioni e le relative applicazioni durante la finestra di cutover pianificata.	Sviluppatore di app, DBA
Convalida la sincronizzazione del database di staging Oracle locale con l'istanza. EC2	Verifica che tutti i messaggi siano stati pubblicati per le attività di replica dall'istanza di SharePlex replica al database di staging Oracle su Amazon EC2 eseguendo alcuni cambi	DBA

Operazione	Description	Competenze richieste
	di registro sul database di origine locale. Per ulteriori informazioni, consulta 6.4.2 Cambio di un file di registro nella documentazione Oracle.	
Convalida la sincronizzazione del database di gestione temporanea di Oracle con il database Amazon RDS for Oracle.	Verifica che tutte le attività di AWS DMS non presentino ritardi né errori, quindi verifica lo stato di convalida delle attività.	DBA
Interrompi la replica di SharePlex Amazon RDS.	Se entrambe le repliche SharePlex e AWS DMS non mostrano errori, interrompi entrambe le repliche.	DBA
Rimappa l'applicazione su Amazon RDS.	Condividi i dettagli dell'endpoint Amazon RDS for Oracle con il server delle applicazioni e le relative applicazioni, quindi avvia l'applicazione per riprendere le operazioni aziendali.	Sviluppatore di app, DBA

Testa l'ambiente di destinazione AWS

Operazione	Description	Competenze richieste
Testa l'ambiente del database di staging Oracle su AWS.	1. Testa la SharePlex replica e verifica che non vi siano lacune di sincronizzazione	SharePlex, Amministrazione Oracle

Operazione	Description	Competenze richieste
	o errori di replica sul database di staging Oracle. 2. Verifica che l'applicazione si comporti come previsto tramite benchmark definiti nell'ambiente locale.	
Testa l'ambiente Amazon RDS.	1. Verifica che tutti i dati propagati su Amazon RDS dopo la replica siano privi di errori. 2. Indirizza un'altra applicazione all'istanza DB di Amazon RDS, quindi esegui test delle prestazioni per verificare il comportamento previsto. Per ulteriori informazioni, consulta Amazon RDS for Oracle nella documentazione di Amazon RDS.	Amministrazione Oracle

Risorse correlate

- Migra con fiducia
- Amazon EC2
- Amazon RDS per Oracle
- AWS Database Migration Service
- Eseguire il debug delle migrazioni AWS DMS: cosa fare quando le cose vanno male (parte 1)
- Eseguire il debug delle migrazioni AWS DMS: cosa fare quando le cose vanno male (parte 2)
- Eseguire il debug delle migrazioni AWS DMS: cosa fare quando le cose vanno male? (Parte 3)

- SharePlex per la replica del database
- SharePlex: replica del database per qualsiasi ambiente

Esegui la migrazione di un database MySQL locale su Amazon EC2

Lorenzo Mota, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni per la migrazione di un database MySQL locale a un database MySQL su un'istanza Amazon Elastic Compute Cloud (Amazon EC2). Il modello illustra l'uso di AWS Database Migration Service (AWS DMS) o di strumenti MySQL nativi come mysqldump per la migrazione. Si concentra sulla migrazione completa del database verso un'istanza DB MySQL.

Il modello è destinato principalmente agli architetti di soluzioni DBAs e soluzioni. Può essere utilizzato in progetti piccoli o grandi, in una fase di test o di migrazione finale. Si consiglia di eseguire almeno un ciclo di test prima di utilizzare questo modello in un ambiente di produzione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database sorgente MySQL in un data center locale

Versioni del prodotto

- MySQL versione 5.5 e successive
- Un sistema operativo di destinazione supportato da Amazon EC2; consulta Amazon EC2 FAQs

Architecture

Stack tecnologico di origine

- Un database MySQL locale

Stack tecnologico Target

- Un'istanza di database MySQL su Amazon EC2

Metodi di migrazione dei dati AWS

- AWS DMS
- Strumenti MySQL nativi come mysqldump o strumenti di terze parti come Percona XtraBackup

Architettura Target

Il diagramma seguente illustra l'implementazione di destinazione di Amazon EC2 dopo il cutover.

Architettura di migrazione dei dati AWS

Utilizzo di AWS DMS:

Il diagramma seguente illustra il flusso di lavoro di migrazione dei dati basato sull'invio AWS DMS di modifiche complete e incrementali al database MySQL di destinazione fino al cutover. La connessione di rete da locale a AWS dipende dai requisiti del client SQL e non rientra nell'ambito di questo modello.

Utilizzando altri strumenti MySQL:

Il diagramma seguente illustra il flusso di lavoro di migrazione dei dati basato sull'utilizzo degli strumenti MySQL per generare file di dump di esportazione dal database locale. Questi file vengono spostati su Amazon Simple Storage Service (Amazon S3) e importati nel database MySQL di destinazione prima del cutover. La connessione di rete da locale a AWS dipende dai requisiti del client SQL e non rientra nell'ambito di questo modello.

Note:

- A seconda delle considerazioni relative ai tempi di inattività e alle dimensioni del database per il cutover finale, puoi utilizzare AWS DMS un altro strumento di acquisizione dei dati di modifica (CDC) per ridurre al minimo i tempi di cutover. Quando si utilizza uno strumento CDC come, ad esempio AWS DMS, è possibile migrare al database di destinazione in pochi minuti.
- Una strategia offline con mysqldump può essere sufficiente se le dimensioni del database e la latenza di rete consentono una breve finestra di migrazione. (Si consiglia di eseguire dei test per ottenere un tempo approssimativo.)
- Di solito una strategia CDC AWS DMS richiede più monitoraggio e complessità rispetto a un'opzione offline.

Tools (Strumenti)

AWS servizi

- AWS Database Migration Service (AWS DMS) supporta diversi database di origine e destinazione. Per informazioni sui database di origine e destinazione MySQL supportati AWS DMS da, vedere Utilizzo di un database compatibile con MySQL come origine e Utilizzo di un database compatibile con MySQL come destinazione AWS DMS per. AWS DMS Se il database di origine non è supportato da, è necessario scegliere un altro metodo per migrare AWS DMS i dati.

Altri strumenti

- mysqldump è un'utilità MySQL che crea un file di dump da un database MySQL per scopi di backup o migrazione.
- Percona XtraBackup è un'utilità open source per eseguire backup non bloccanti su database MySQL.

Epiche

Pianifica la migrazione

Operazione	Description	Competenze richieste
Convalida le versioni del database.	Convalida le versioni dei database di origine e di destinazione. Per informazioni sulle versioni di MySQL supportate AWS DMS da, consulta Sources AWS DMS for e Targets AWS DMS nella documentazione. AWS DMS	DBA
Identifica il sistema operativo di destinazione.	Determina la versione del sistema operativo di destinazione. Per un elenco dei sistemi operativi di destinazione	DBA, amministratore di sistema

Operazione	Description	Competenze richieste
	supportati da Amazon EC2, consulta Amazon EC2. FAQs	
Identifica i requisiti hardware.	Determina i requisiti hardware per l'istanza del server di destinazione in base all'elenco di compatibilità MySQL e ai requisiti di capacità.	DBA, amministratore di sistema
Identifica i requisiti di archiviazione.	Determina il tipo e la capacità di archiviazione per il database di destinazione.	DBA, amministratore di sistema
Identifica i requisiti di rete.	Determina i requisiti di rete come latenza e larghezza di banda.	DBA, amministratore di sistema
Scegli il tipo di istanza di destinazione.	Scegli il tipo di istanza di destinazione in base alla capacità, alle funzionalità di archiviazione e alle funzionalità di rete.	DBA, amministratore di sistema
Identifica i requisiti di sicurezza.	Determina i requisiti di sicurezza di accesso alla rete o all'host per i database di origine e di destinazione.	DBA, amministratore di sistema
Identifica gli utenti.	Determina l'elenco degli utenti del sistema operativo per l'installazione del software MySQL. Per ulteriori informazioni, consulta la documentazione MySQL.	DBA, amministratore di sistema
Determinare una strategia di backup.		DBA

Operazione	Description	Competenze richieste
Determina i requisiti di disponibilità.		DBA
Identifica la strategia di migrazione o commutazione delle applicazioni.		DBA, amministratore di sistema

Configura l'infrastruttura

Operazione	Description	Competenze richieste
Crea un cloud privato virtuale (VPC) e sottoreti.	Configura tabelle di routing, gateway Internet, gateway NAT e sottoreti. Per ulteriori informazioni, consulta le opzioni di configurazione VPC nella documentazione di Amazon VPC.	Amministratore di sistema
Crea gruppi di sicurezza e liste di controllo degli accessi alla rete (ACLs).	Configura le porte (l'impost azione predefinita per MySQL è 3306) e gli intervalli CIDR o IPs specifici a seconda delle tue esigenze.	Amministratore di sistema
Configura e avvia un'istanza EC2.	Per istruzioni, consulta Avvio di un'istanza EC2 nella documentazione di Amazon EC2.	Amministratore di sistema

Installa il software MySQL

Operazione	Description	Competenze richieste
Creare utenti e gruppi.	Crea gli utenti e i gruppi del sistema operativo che devono accedere al server e al database. Per ulteriori informazioni, consulta Access Control and Account Management nella documentazione MySQL.	DBA, amministratore di sistema
Scarica MySQL.	Scarica il software MySQL. Per istruzioni e file binari, consulta Installazione di MySQL nella documentazione di MySQL.	DBA, amministratore di sistema
Installa MySQL sull'istanza EC2 e configura il server.	Connect alla tua istanza EC2 e installa il software MySQL. Per ulteriori informazioni, consulta Connect to your EC2 istance (Connect to your EC2) nella documentazione di Amazon EC2.	DBA, amministratore di sistema

Migrazione dei dati: opzione 1

Operazione	Description	Competenze richieste
Migra i dati con strumenti nativi MySQL o di terze parti.	Questa opzione utilizza strumenti MySQL nativi o strumenti di terze parti per migrare oggetti e dati del database. Per istruzioni, consulta la documentazione	DBA

Operazione	Description	Competenze richieste
	per mysqldump o Percona (per la migrazione fisica). XtraBackup Per ulteriori informazioni sull'utilizzo di questi strumenti, consulta il post di AWS blog Opzioni di migrazione per MySQL ad Amazon RDS for MySQL o Amazon Aurora MySQL.	

Migrazione dei dati — opzione 2

Operazione	Description	Competenze richieste
Esegui la migrazione dei dati con AWS DMS.	Per ulteriori informazioni, consulta Visualizzazione ad alto livello della AWS DMS AWS DMS documentazione.	DBA

Preparatevi per il cutover

Operazione	Description	Competenze richieste
Raccogli il numero di oggetti.	Raccogli il numero di oggetti dal database di origine e dai nuovi database di destinazione. Correggi eventuali discrepanze nel database di destinazione.	DBA
Controlla le dipendenze.	Verifica che le dipendenze (collegamenti) da e verso altri database siano ancora valide e funzionino correttamente.	DBA

Operazione	Description	Competenze richieste
Prova.	Se si tratta di un ciclo di test, esegui test delle query, raccogli le metriche e risolvi eventuali problemi.	DBA

Tagliare

Operazione	Description	Competenze richieste
Sposta i clienti.	Trasferisci i client applicativi alla nuova infrastruttura.	DBA, proprietario dell'app, amministratore di sistema
Fornire supporto.	Fornire supporto durante i test funzionali delle applicazioni.	DBA

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse.	Chiudi l'istanza di AWS DMS replica e le altre AWS risorse temporanee.	DBA, amministratore di sistema
Documenti di revisione e progetto.	Rivedi e convalida i documenti del progetto.	DBA, proprietario dell'app, amministratore di sistema
Raccogli metriche.	Raccogli parametri come il tempo necessario per la migrazione, la percentuale di modifiche manuali rispetto alle modifiche assistite da strumenti e i risparmi sui costi.	DBA, proprietario dell'app, amministratore di sistema

Operazione	Description	Competenze richieste
Chiudi il progetto.	Chiudi il progetto di migrazion e e fornisci feedback.	DBA, proprietario dell'app, amministratore di sistema
Disattivazione del database di origine.	Disattiva il database MySQL locale.	DBA, amministratore di sistema

Risorse correlate

Riferimenti

- Documentazione Amazon EC2
- AWS DMS documentazione
- Prezzi di Amazon EC2
- AWS DMS Step-by-Step Procedure dettagliate
- mysqldump
- Percona XtraBackup

Tutorial e video

- Guida introduttiva con AWS DMS
- Introduzione ad Amazon EC2: server e hosting cloud elastici con AWS(video)

Monitora Amazon Aurora per le istanze senza crittografia

Mansi Suratwala, Amazon Web Services

Riepilogo

Questo modello fornisce un CloudFormation modello Amazon Web Services (AWS) che puoi implementare per configurare notifiche automatiche quando un'istanza Amazon Aurora viene creata senza la crittografia attivata.

Aurora è un motore di database relazionale completamente gestito compatibile con MySQL e PostgreSQL. Con alcuni carichi di lavoro, Aurora assicura un throughput fino a cinque volte superiore a MySQL e fino al triplo di PostgreSQL e non richiede alcuna modifica alla maggior parte delle applicazioni esistenti.

Il CloudFormation modello crea un evento Amazon CloudWatch Events e una funzione AWS Lambda. L'evento utilizza AWS CloudTrail per monitorare la creazione di qualsiasi istanza Aurora o il ripristino point-in-time di un'istanza esistente. L'evento Cloudwatch Events avvia la funzione Lambda, che verifica se la crittografia è abilitata. Se la crittografia non è attivata, la funzione Lambda invia una notifica Amazon Simple Notification Service (Amazon SNS) che ti informa della violazione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo

Limitazioni

- Questo controllo del servizio funziona solo con le istanze di Amazon Aurora. Non supporta altre istanze di Amazon Relational Database Service (Amazon RDS).
- Il CloudFormation modello deve essere distribuito solo per **`CreateDBInstance`** e. **`RestoreDBClusterToPointInTim`**

Versioni del prodotto

- Versioni PostgreSQL supportate in Amazon Aurora
- Versioni di MySQL supportate in Amazon Aurora

Architecture

Stack tecnologico Target

- Amazon Aurora
- AWS CloudTrail
- Amazon CloudWatch
- AWS Lambda
- Amazon Simple Storage Service (Amazon S3)
- Amazon SNS

Architettura Target

Automazione e scalabilità

Puoi utilizzare il CloudFormation modello più volte per diverse regioni e account. Devi eseguirlo solo una volta in ogni regione o account.

Tools (Strumenti)

Strumenti

- Amazon Aurora — Amazon Aurora è un motore di database relazionale completamente gestito compatibile con MySQL e PostgreSQL.
- AWS CloudTrail: AWS ti CloudTrail aiuta a gestire la governance, la conformità e il controllo operativo e dei rischi del tuo account AWS. Le azioni intraprese da un utente, un ruolo o un servizio AWS vengono registrate come eventi in CloudTrail.
- Amazon CloudWatch Events: Amazon CloudWatch Events offre un near-real-time flusso di eventi di sistema che descrivono i cambiamenti nelle risorse AWS.
- AWS Lambda: AWS Lambda è un servizio di elaborazione che supporta l'esecuzione di codice senza effettuare il provisioning o la gestione di server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti altamente scalabile che puoi utilizzare per un'ampia gamma di soluzioni di storage, tra cui siti Web, applicazioni mobili, backup e data lake.

- Amazon SNS — Amazon Simple Notification Service (Amazon SNS) è un servizio gestito che fornisce il recapito di messaggi tramite Lambda, HTTP, e-mail, notifiche push mobili e messaggi di testo mobili (SMS).

Codice

Un file.zip del progetto è disponibile come allegato.

Epiche

Crea il bucket S3 per lo script Lambda

Operazione	Description	Competenze richieste
Definisci il bucket S3.	Apri la console Amazon S3 e scegli o crea un bucket S3. Questo bucket S3 ospiterà il file.zip con codice Lambda. Il tuo bucket S3 deve trovarsi nella stessa regione di Aurora. Il nome del bucket S3 non può contenere barre iniziali.	Architetto del cloud

Carica il codice Lambda nel bucket S3

Operazione	Description	Competenze richieste
Carica il codice Lambda.	Carica il file.zip con codice Lambda fornito nella sezione Allegati nel bucket S3 che hai definito.	Architetto del cloud

Implementa il modello CloudFormation

Operazione	Description	Competenze richieste
Implementa il CloudFormation modello.	Sulla CloudFormation console, distribuisci il `RDS_Aurora_Encryption_At_Rest.yml` CloudFormation modello fornito come allegato a questo modello. Nella prossima epopea, fornisci i valori per i parametri del modello.	Architetto del cloud

Completa i parametri nel CloudFormation modello

Operazione	Description	Competenze richieste
Fornisci il nome del bucket S3.	Inserisci il nome del bucket S3 che hai creato o scelto nella prima epic.	Architetto del cloud
Fornisci la chiave S3.	Fornisci la posizione del file.zip del codice Lambda nel tuo bucket S3, senza barre iniziali (ad esempio,) . `<directory>/<file-name>.zip`	Architetto del cloud
Fornisci un indirizzo email.	Fornisci un indirizzo e-mail attivo per ricevere le notifiche di Amazon SNS.	Architetto del cloud
Definisci il livello di registrazione.	Definisci il livello e la frequenza di registrazione per la tua funzione Lambda. `Info`indica messaggi informati	Architetto del cloud

Operazione	Description	Competenze richieste
	vi dettagliati sullo stato di avanzamento dell'appl icazione. `Error`indica eventi di errore che potrebbero comunque consentire all'appli cazione di continuare a funzionare. `Warning`indica situazioni potenzialmente dannose.	

Confermare la sottoscrizione

Operazione	Description	Competenze richieste
Confermare la sottoscrizione.	Quando il modello viene distribuito correttamente, invia un messaggio e-mail di sottoscrizione all'indirizzo e-mail fornito. Per ricevere notifiche, è necessario confermare questa sottoscri zione e-mail.	Architetto del cloud

Risorse correlate

- Creazione di un bucket S3
- Caricamento di file su un bucket S3
- Creazione di un cluster database Amazon Aurora
- Creazione di una regola CloudWatch Events che si attiva su una chiamata API AWS utilizzando AWS CloudTrail

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Monitora GoldenGate i log di Oracle utilizzando Amazon CloudWatch

Chithra Krishnamurthy, Amazon Web Services

Riepilogo

Oracle GoldenGate fornisce la replica in tempo reale tra Amazon Relational Database Service (Amazon RDS) per database Oracle o tra database Oracle ospitati su Amazon Elastic Compute Cloud (Amazon EC2). Supporta la replica sia unidirezionale che bidirezionale.

Quando si utilizza GoldenGate per la replica, il monitoraggio è fondamentale per verificare che il GoldenGate processo sia attivo e funzionante e per assicurarsi che i database di origine e di destinazione siano sincronizzati.

Questo modello spiega i passaggi per implementare il CloudWatch monitoraggio di Amazon per un log degli GoldenGate errori e come impostare allarmi per inviare notifiche per eventi specifici, ad esempio intraprendere le azioni appropriate per riprendere rapidamente la replica. `STOP ABEND`

Prerequisiti e limitazioni

Prerequisiti

- GoldenGate installato e configurato su un'istanza EC2, in modo da poter configurare il CloudWatch monitoraggio su tali istanze EC2. Se desideri monitorare la replica bidirezionale in tutte le GoldenGate regioni AWS, devi installare l' CloudWatch agente in ogni istanza EC2 in cui è in esecuzione il GoldenGate processo.

Limitazioni

- Questo schema spiega come monitorare il GoldenGate processo utilizzando CloudWatch. CloudWatch non monitora il ritardo di replica o i problemi di sincronizzazione dei dati durante la replica. È necessario eseguire query SQL separate per monitorare il ritardo di replica o gli errori relativi ai dati, come spiegato nella documentazione. GoldenGate

Versioni del prodotto

- Questo documento si basa sull'implementazione di Oracle GoldenGate 19.1.0.0.4 per Oracle su Linux x86-64. Tuttavia, questa soluzione è applicabile a tutte le versioni principali di. GoldenGate

Architecture

Stack tecnologico Target

- GoldenGate binari per Oracle installati su un'istanza EC2
- Amazon CloudWatch
- Amazon Simple Notification Service (Amazon SNS)

Architettura Target

Tools (Strumenti)

Servizi AWS

- Amazon CloudWatch è un servizio di monitoraggio che viene utilizzato in questo schema per monitorare i log degli GoldenGate errori.
- Amazon SNS è un servizio di notifica dei messaggi utilizzato in questo modello per inviare notifiche e-mail.

Altri strumenti

- Oracle GoldenGate è uno strumento di replica dei dati che puoi utilizzare per i database Amazon RDS for Oracle o per i database Oracle ospitati su Amazon EC2.

Fasi di implementazione di alto livello

1. Crea un ruolo AWS Identity and Access Management (IAM) per l' CloudWatch agente.
2. Collega il ruolo IAM all'istanza EC2 in cui vengono generati i log degli GoldenGate errori.
3. Installa l' CloudWatch agente sull'istanza EC2.
4. Configura i file di configurazione CloudWatch dell'agente: `awscli.conf` `eawslogs.conf`.
5. Avvia l' CloudWatch agente.
6. Crea filtri metrici nel gruppo di log.
7. Configura Amazon SNS.
8. Crea un allarme per i filtri metrici. Amazon SNS invia avvisi e-mail quando tali filtri rilevano eventi.

Per istruzioni dettagliate, consulta la sezione successiva.

Epiche

Passaggio 1. Crea un ruolo IAM per l'agente CloudWatch

Operazione	Description	Competenze richieste
Creare il ruolo IAM.	L'accesso alle risorse AWS richiede autorizzazioni, quindi puoi creare ruoli IAM per includere le autorizzazioni necessarie a ciascun server per eseguire l' CloudWatch agente. Per creare il ruolo IAM: 1. Accedi alla Console di gestione AWS, quindi apri la console di IAM all'indirizzo https://console.aws.amazon.com/iam/. 2. Nel pannello di navigazione, scegli Ruoli e quindi Crea ruolo. 3. Per il tipo di entità affidabile, scegli il servizio AWS. 4. Per i casi d'uso comuni, scegli EC2, quindi scegli Avanti. 5. Nell'elenco delle politiche, seleziona la casella di controllo accanto a CloudWatchAgentServerPolicy. Se necessari	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	o, utilizzare la casella di ricerca per trovare la policy. 6. Scegli Next (Successivo). 7. Per Role name (Nome ruolo), inserire un nome per il nuovo ruolo, ad esempio `goldengate-cw-monitoring-role` o un altro nome che preferisci. 8. (Facoltativo) Per Role Description (Descrizione ruolo), immetti una descrizione. 9. Conferma che sia CloudWatchAgentServerPolicyvisualizzato sotto Nome della politica. 10. (Facoltativo) Aggiungi uno o più tag (coppie chiave-valore) per organizzare, tracciare o controllare l'accesso per questo ruolo, quindi scegli Crea ruolo.	

Passaggio 2. Collega il ruolo IAM all' GoldenGate istanza EC2

Operazione	Description	Competenze richieste
Collega il ruolo IAM all'istanza EC2 in cui vengono generati i log degli GoldenGate errori.	I log degli errori generati da GoldenGate devono essere compilati CloudWatch e monitorati, quindi è necessario collegare il ruolo IAM	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	creato nella fase 1 all'istanza EC2 in cui è in esecuzione. GoldenGate Per associare un ruolo IAM a un'istanza: 1. Apri la console Amazon EC2 all'indirizzo https://console.aws.amazon.com/ec2/. 2. Nel riquadro di navigazione, scegli Istanze, quindi trova l'istanza in cui GoldenGate è in esecuzione. 3. Seleziona l'istanza, quindi scegli il ruolo Azioni, Sicurezza, Modifica IAM. 4. Seleziona il ruolo IAM creato nel primo passaggio da collegare all'istanza, quindi scegli Salva.	

Fasi 3-5. Installa e configura l' CloudWatch agente sull'istanza Goldengate EC2

Operazione	Description	Competenze richieste
Installa l' CloudWatch agente sull'istanza GoldenGate EC2.	Per installare l'agente, esegui il comando: `sudo yum install -y awslogs`	Informazioni generali su AWS

Operazione	Description	Competenze richieste

Modifica i file di configurazione dell'agente.	1. Eseguire il seguente comando seguente.
	`sudo su -`
	2. Modifica questo file per aggiornare la regione AWS, se necessario.
	`cat /etc/awslogs/conf`
	`[plugins]`
	`cwlogs = cwlogs`
	`[default]`
	`region = us-east-1`
	3. Modifica il `/etc/awslogs/awslogs.conf` file per aggiornare il nome del file, il nome del gruppo di log e il date/time formato. È necessario specificare il formato in cui deve corrispondere date/time alla data`ggserror.log` ; in caso contrario, il flusso di log non verrà visualizzato CloudWatch. Esempio:
	`datetime_format = %Y-%m-%dT%H:%M:%S%z`
	`file = /u03/oracle/oragg/ggserr.log`
	`log_group_name = goldengate_monitor`
		Informazioni generali su AWS

Operazione	Description	Competenze richieste
Avvia l' CloudWatch agente.	Per avviare l'agente, utilizzare il seguente comando. `$ sudo service awslogsd start` Dopo aver avviato l'agente, è possibile visualizzare il gruppo di log nella CloudWatch console. Il flusso di log conterrà il contenuto del file.	Informazioni generali su AWS

Fase 6. Crea filtri metrici per il gruppo di log

Operazione	Description	Competenze richieste
Crea filtri metrici per le parole chiave ABEND e STOPPED.	Quando crei filtri metrici per il gruppo di log, ogni volta che i filtri vengono identificati nel log degli errori, avvia un allarme e invia una notifica e-mail basata sulla configurazione di Amazon SNS. Per creare filtri metrici: 1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/. 2. Scegli il nome del gruppo di log.	CloudWatch

Operazione	Description	Competenze richieste
	3. Scegli Actions (Operazioni) e quindi Create metric filter (Crea filtro parametri). 4. Per il pattern di filtro, specificate uno schema comeABEND. 5. Scegli Successivo e poi inserisci un nome per il filtro. 6. In Dettagli metrici, per Metric namespace, inserisci un nome per lo spazio dei CloudWatch nomi in cui verrà pubblicata la metrica. Se questo spazio dei nomi non esiste già, assicurati che sia selezionato Crea nuovo. 7. Per Valore metrico, inserisci 1, perché il filtro metrico conta le occorrenze delle parole chiave nel filtro. 8. Imposta l'unità su Nessuno. 9. Scegli Crea filtro parametri . Puoi trovare il filtro dei parametri che hai creato nel riquadro di navigazione. 10. Crea un altro filtro metrico per il ST0PPED pattern. All'interno di un gruppo di log, puoi creare più filtri metrici e impostare gli allarmi singolarmente.	

Fase 7. Configurazione di Amazon SNS

Operazione	Description	Competenze richieste
Creare un argomento SNS.	In questo passaggio, configuri Amazon SNS per creare allarmi per i filtri metrici. Per creare un argomento SNS: 1. Accedi alla console Amazon SNS da casahttps://console.aws.amazon.com/sns/. 2. Nella casella Crea argomento, inserisci il nome di un argomento `goldengate-alert` , ad esempio, quindi scegli Passaggio successivo. 3. Per Tipo, scegliere Standard. 4. Vai in fondo al modulo e scegli Creare un argomento . La console apre la pagina nuovi argomenti Dettagli.	Amazon SNS
Crea un abbonamento.	Per creare un abbonamento all'argomento: 1. Nel pannello di navigazione sinistro scegli Sottoscrizioni. 2. Nella pagina Sottoscrizioni scegli Crea sottoscrizione. 3. Nella pagina Crea abbonamento, scegli il campo Arn dell'argomento	Amazon SNS

Operazione	Description	Competenze richieste
	per visualizzare un elenco degli argomenti del tuo account AWS. 4. Scegliere laargomento creato nei passaggi precedenti. 5. Per Protocol, scegliere Email. 6. In Endpoint immetti l'indirizzo e-mail utilizzabile per ricevere le notifiche. 7. Scegli Crea abbonamento. La console apre la pagina dei dettagli del nuovo abbonamento. 8. Controlla la tua casella di posta elettronica per ricevere un messaggio da AWS Notifications, quindi scegli Conferma abbonamento nell'e-mail. Amazon SNS apre il browser Web e visualizza una conferma dell'abbonamento con il tuo ID abbonamento.	

Fase 8. Crea un allarme per inviare notifiche per i filtri metrici

Operazione	Description	Competenze richieste
Crea un allarme per l'argomento SNS.	Per creare un allarme basato su un filtro metrico per gruppi di log: 1. Apri la console all' CloudWatch indirizzo. https://console.aws.amazon.com/cloudwatch/ 2. Nel pannello di navigazione scegli Logs (Log), quindi Log groups (Gruppi di log). 3. Scegli il gruppo di log che include il filtro parametri. 4. Scegli Metric filters (Filtri parametri). 5. Nella scheda Filtri metrici, seleziona la casella di controllo relativa al filtro metrico su cui vuoi basare l'allarme. 6. Scegli Crea allarme. 7. Per Condizioni, specifica quanto segue in ogni sezione: • For Threshold type (Tipo di soglia), scegli Static (Statica). • Per Whenever is.. <metric-name> , scegli Maggiore. • Per di... , specificare 0.	CloudWatch

Operazione	Description	Competenze richieste
	8. Scegli Next (Successivo). 9. In Notifica: • Per Alarm state trigger (Attivazione stato allarme), scegli In alarm (In allarme). • Per Invia notifica al seguente argomento SNS, scegli Seleziona un argomento esistente. • Nella casella e-mail, seleziona l'argomento Amazon SNS che hai creato nel passaggio precedente. 10. Scegli Next (Successivo). 11. Per Name and description (Nome e descrizione), inserisci un nome e una descrizione per il tuo allarme. **Note** Per la descrizione, puoi specificare il nome dell'istanza in modo che l'e-mail di notifica sia descrittiva. 12. Per Anteprima e creazione, verifica che la configura	

Operazione	Description	Competenze richieste
	zione sia corretta, quindi scegli Crea allarme. Dopo questi passaggi, ogni volta che questi schemi vengono rilevati nel file di registro degli GoldenGate errori (`ggserr.log`) che stai monitorando, riceverai una notifica via email.	

Risoluzione dei problemi

Problema	Soluzione
Il flusso di log proveniente dal registro GoldenGate degli errori non confluisce in CloudWatch.	Controlla il `/etc/awslogs/awslogs.conf` file per verificare il nome del file, il nome del gruppo di log e il date/time formato. È necessario specificare il date/time formato della data in cui deve corrispondere`ggserror.log` . In caso contrario, il flusso di log non confluirà in CloudWatch.

Risorse correlate

- CloudWatch Documentazione Amazon
- Raccolta di metriche e registri con l'agente CloudWatch
- Documentazione Amazon SNS

Ripiattaforma Oracle Database Enterprise Edition alla Standard Edition 2 su Amazon RDS per Oracle

Showunmi Lanre (Lan-Ray) e Tarun Chawla, Amazon Web Services

Riepilogo

Oracle Database Enterprise Edition (EE) è una scelta popolare per l'esecuzione di applicazioni in molte aziende. In alcuni casi, tuttavia, le applicazioni utilizzano poche o nessuna funzionalità di Oracle Database EE, quindi non esiste alcuna giustificazione per sostenere ingenti costi di licenza. Puoi ottenere risparmi sui costi effettuando il downgrade di tali database a Oracle Database Standard Edition 2 (SE2) quando esegui la migrazione ad Amazon RDS.

Questo modello descrive come effettuare il downgrade da Oracle Database EE a Oracle Database SE2 durante la migrazione da locale ad Amazon RDS for Oracle. I passaggi presentati in questo modello si applicano anche se il database EE Oracle è già in esecuzione su Amazon RDS o su un'istanza Amazon Elastic Compute Cloud EC2 (Amazon).

Per ulteriori informazioni, consulta la guida AWS Prescriptive Guidance su come valutare il downgrade dei database Oracle alla Standard Edition 2 su AWS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Edizione Enterprise del database Oracle
- Uno strumento client, come Oracle SQL Developer o SQL*Plus, per la connessione e l'esecuzione di comandi SQL sul database Oracle
- Utente del database per l'esecuzione della valutazione; ad esempio, uno dei seguenti:
  - Utente con privilegi sufficienti per eseguire la valutazione di AWS Schema Conversion Tool (AWS SCT)
  - Utente con privilegi sufficienti per eseguire query SQL sulle tabelle dei dizionari del database Oracle
- Utente del database per l'esecuzione della migrazione del database; ad esempio, uno dei seguenti:
  - Utente con privilegi sufficienti per eseguire AWS Database Migration Service (AWS DMS)
  - Utente con privilegi sufficienti per eseguire l'esportazione e l'importazione di Oracle Data Pump

- Utente con privilegi sufficienti per eseguire Oracle GoldenGate

Limitazioni

- Amazon RDS for Oracle ha una dimensione massima del database. Per ulteriori informazioni, consulta Storage delle istanze di database Amazon RDS.

Versioni del prodotto

La logica generale descritta in questo documento si applica alle versioni Oracle dalla 9i in poi. Per le versioni supportate dei database autogestiti e Amazon RDS for Oracle, consulta la documentazione di AWS DMS.

Per identificare l'utilizzo delle funzionalità nei casi in cui AWS SCT non è supportato, esegui query SQL sul database di origine. Per migrare da versioni precedenti di Oracle in cui AWS DMS e Oracle Data Pump non sono supportati, utilizza le utilità Oracle Export and Import.

Per un elenco aggiornato delle versioni ed edizioni supportate, consulta Oracle on Amazon RDS nella documentazione AWS. Per informazioni dettagliate sui prezzi e sulle classi di istanze supportate, consulta Prezzi di Amazon RDS per Oracle.

Architecture

Stack tecnologico di origine

- Oracle Database Enterprise Edition in esecuzione in locale o su Amazon EC2

Scegli lo stack tecnologico utilizzando strumenti Oracle nativi

- Amazon RDS per Oracle che esegue Oracle Database SE2

1. Esporta i dati utilizzando Oracle Data Pump.
2. Copia i file di dump su Amazon RDS tramite un collegamento al database.
3. Importa file di dump in Amazon RDS utilizzando Oracle Data Pump.

Stack tecnologico di destinazione con AWS DMS

- Amazon RDS per Oracle che esegue Oracle Database SE2
- AWS DMS

1. Esporta i dati utilizzando Oracle Data Pump con FLASHBACK_SCN.
2. Copia i file di dump su Amazon RDS tramite un collegamento al database.
3. Importa file di dump in Amazon RDS utilizzando Oracle Data Pump.
4. Usa AWS DMS Change Data Capture (CDC).

Tools (Strumenti)

Servizi AWS

- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS. Questo modello utilizza Amazon RDS for Oracle.
- AWS SCT fornisce un'interfaccia utente basata su progetti per valutare, convertire e copiare automaticamente lo schema del database Oracle di origine in un formato compatibile con Amazon RDS for Oracle. AWS SCT consente di analizzare i potenziali risparmi sui costi che è possibile ottenere cambiando il tipo di licenza da Enterprise a Standard Edition di Oracle. La sezione License Evaluation and Cloud Support del report AWS SCT fornisce informazioni dettagliate sulle funzionalità Oracle in uso in modo da poter prendere una decisione informata durante la migrazione ad Amazon RDS for Oracle.

Altri strumenti

- Le utilità native di importazione ed esportazione di Oracle supportano lo spostamento dei dati Oracle all'interno e all'esterno dei database Oracle. Oracle offre due tipi di utilità di importazione ed esportazione dei database: Original Export and Import (per le release precedenti) e Oracle Data Pump Export and Import (disponibile in Oracle Database 10g release 1 e successive).

- Oracle GoldenGate offre funzionalità di replica in tempo reale che consentono di sincronizzare il database di destinazione dopo un caricamento iniziale. Questa opzione può aiutare a ridurre i tempi di inattività delle applicazioni durante il go-live.

Epiche

Effettua una valutazione pre-migrazione

Operazione	Description	Competenze richieste
Convalida i requisiti del database per le tue applicazioni.	Assicurati che le tue applicazioni siano certificate per l'esecuzione su Oracle Database SE2. Consulta direttamente la documentazione del fornitore del software, dello sviluppatore o dell'applicazione.	Sviluppatore di app, DBA, proprietario dell'app
Esamina l'uso delle funzionalità EE direttamente nel database.	Per determinare l'utilizzo della funzionalità EE, effettuate una delle seguenti operazioni: • Genera un rapporto di valutazione AWS SCT per il tuo database Oracle EE. Il rapporto indica quali funzionalità dal tuo attuale database EE devono essere rimosse se desideri modificare i tipi di licenza. • Se disponi di un account Oracle Support, ottieni ed esegui lo script `options_packs_usage_statistics.sql` nel documento di supporto 1317265.1 per	Proprietario dell'app, DBA, sviluppatore dell'app

Operazione	Description	Competenze richieste
	generare un rapporto sulle opzioni e le funzionalità utilizzate nel tuo database Oracle. • Interroga DBA_FEATURE_USAGE_STATISTICS per visualizzare i dettagli di tutte le funzionalità in uso.	
Identifica l'uso delle funzionalità EE per le attività operative.	Gli amministratori di database o applicazioni a volte si affidano a funzionalità esclusive di EE per le attività operative. Gli esempi più comuni includono le attività di manutenzione online (ricostruzione dell'indice, spostamento delle tabelle) e l'uso del parallelismo nei processi in batch. Queste dipendenze possono essere mitigate modificando le operazioni ove possibile. Identifica l'uso di queste funzionalità e prendi una decisione in base ai costi rispetto ai vantaggi. Utilizza la tabella di confronto tra Oracle Database EE e le SE2 funzionalità come guida per identificare le funzionalità disponibili in Oracle Database SE2.	Sviluppatore di app, DBA, proprietario dell'app

Operazione	Description	Competenze richieste
Esamina i modelli di carico di lavoro del database EE Oracle.	Oracle Database limita SE2 automaticamente l'utilizzo a un massimo di 16 thread della CPU in qualsiasi momento. Se il database Oracle EE è autorizzato a utilizzare Oracle Diagnostic Pack, utilizza lo strumento Automatic Workload Repository (AWR) o le viste DBA_HIST_* per analizzare i modelli di carico di lavoro del database e determinare se il limite massimo di 16 thread della CPU avrà un impatto negativo sui livelli di servizio in caso di downgrade a. SE2 Assicurati che la valutazione copra i periodi di picco di attività, ad esempio l'elaborazione di fine giornata, mese o anno.	Proprietario dell'app, DBA, sviluppatore dell'app

Prepara l'infrastruttura di destinazione su AWS

Operazione	Description	Competenze richieste
Implementa e configura l'infrastruttura di rete.	Crea un cloud privato virtuale (VPC) e sottoreti, gruppi di sicurezza ed elenchi di controllo degli accessi alla rete.	Amministratore AWS, architetto cloud, amministratore di rete, DevOps ingegnere

Operazione	Description	Competenze richieste
Effettua il provisioning del database Amazon RDS for SE2 Oracle.	Effettua il provisioning del database Amazon RDS for SE2 Oracle di destinazione per soddisfare i requisiti di prestazioni, disponibilità e sicurezza delle tue applicazioni. Consigliamo la configurazione Multi-AZ per i carichi di lavoro di produzione. Tuttavia, per migliorare le prestazioni di migrazione, puoi rimandare l'attivazione di Multi-AZ a dopo la migrazione dei dati.	Amministratore cloud, architetto cloud, DBA, DevOps ingegnere, amministratore AWS
Personalizza l'ambiente Amazon RDS.	Configura parametri e opzioni personalizzati e abilita un monitoraggio aggiuntivo. Per ulteriori informazioni, consulta Best practice per la migrazione ad Amazon RDS for Oracle.	Amministratore AWS, amministratore di sistema AWS, amministratore cloud, DBA, architetto cloud

Esegui la migrazione, il dry run e il test delle applicazioni

Operazione	Description	Competenze richieste
Migrazione dei dati (funzionamento a secco).	Esegui la migrazione dei dati dal database Oracle EE di origine all'istanza di database Amazon RDS for SE2 Oracle utilizzando l'approccio più adatto al tuo ambiente specifico. Seleziona una strategia di migrazione basata su fattori quali	DBA

Operazione	Description	Competenze richieste
	dimensioni, complessità e periodo di inattività disponibile. Utilizza uno o una combinazi one dei seguenti: • Strumenti Oracle nativi come Oracle Data Pump (consigliato), le utilità Oracle Import-Export e Oracle. GoldenGate • AWS DMS, che utilizza il pieno carico con replica continua tramite CDC.	
Convalida il database di destinazione.	Esegui la convalida post-migrazione dello storage del database e degli oggetti di codice. Esamina i log di migrazione e risolvi eventuali problemi identificati. Per ulteriori informazioni, consulta la guida Migrazione dei database Oracle al cloud AWS.	DBA

Operazione	Description	Competenze richieste
Prova le applicazioni.	Gli amministratori delle applicazioni e dei database devono eseguire test funzionali, prestazionali e operativi, a seconda dei casi. Per ulteriori informazioni, consulta Best practice per la migrazione ad Amazon RDS for Oracle. Infine, ottieni l'approvazione dei risultati dei test dalle parti interessate.	Sviluppatore di app, proprietario dell'app, DBA, ingegnere addetto alla migrazione, responsabile della migrazione

Tagliare

Operazione	Description	Competenze richieste
Aggiorna i dati da Oracle Database EE.	Seleziona un approccio di aggiornamento dei dati in base ai requisiti di disponibilità dell'applicazione. Per ulteriori informazioni, consulta i metodi di migrazione in Strategie per la migrazione dei database Oracle su AWS. Ad esempio, puoi ottenere tempi di inattività vicini allo zero utilizzando strumenti come Oracle o GoldenGate e AWS DMS con replica continua. Se il periodo di inattività lo consente, puoi eseguire il cutover finale dei dati utilizzando metodi offline	Proprietario dell'app, responsabile Cutover, DBA, ingegnere addetto alla migrazione, responsabile della migrazione

Operazione	Description	Competenze richieste
	come le utilità Oracle Data Pump o Original Export-Import.	
Indirizza le applicazioni all'istanza del database di destinazione.	Aggiorna i parametri di connessione nelle applicazioni e in altri client in modo che puntino al database Amazon RDS for SE2 Oracle.	Sviluppatore di app, proprietario dell'app, tecnico addetto alla migrazione, responsabile della migrazione, responsabile Cutover
Esegui attività post-migrazione.	Esegui attività successive alla migrazione dei dati, come l'attivazione di Multi-AZ, la convalida dei dati e altri controlli.	DBA, ingegnere addetto alla migrazione
Esegui il monitoraggio post-cutover.	Utilizza strumenti come Amazon CloudWatch e Amazon RDS Performance Insights per monitorare il database Amazon RDS for SE2 Oracle.	Sviluppatore di app, proprietario dell'app, amministratore AWS, DBA, ingegnere addetto alla migrazione

Risorse correlate

Prontuario AWS

- Migrazione dei database Oracle al cloud AWS (guida)
- Valuta il downgrade dei database Oracle alla Standard Edition 2 su AWS (guida)
- Migrazione di un database Oracle locale su Amazon RDS for Oracle (modello)
- Migrazione di un database Oracle on-premise su Amazon RDS per Oracle utilizzando Oracle Data Pump (modello)

Post di blog

- Migrazione dei database Oracle con tempi di inattività quasi nulli utilizzando AWS DMS
- Analisi della gestione delle prestazioni in Oracle SE con Amazon RDS for Oracle
- Gestione del piano SQL in Oracle SE con Amazon RDS for Oracle
- Implementazione del partizionamento delle tabelle in Oracle Standard Edition: parte 1

Replica i database mainframe su AWS utilizzando Precisly Connect

Lucio Pereira, Sayantan Giri e Balaji Mohan, Amazon Web Services

Riepilogo

Questo modello descrive i passaggi per replicare i dati dai database mainframe agli archivi dati Amazon quasi in tempo reale utilizzando Precisly Connect. Implementa un'architettura basata su eventi con Amazon Managed Streaming for Apache Kafka (Amazon MSK) e connettori di database personalizzati nel cloud per migliorare scalabilità, resilienza e prestazioni.

Precisly Connect è uno strumento di replica che acquisisce i dati dai sistemi mainframe legacy e li integra in ambienti cloud. I dati vengono replicati dai mainframe ad AWS tramite Change Data Capture (CDC) utilizzando flussi di messaggi quasi in tempo reale con pipeline di dati eterogenee a bassa latenza e ad alto throughput.

Questo modello copre anche una strategia di disaster recovery per pipeline di dati resilienti con replica dei dati in più regioni e routing di failover.

Prerequisiti e limitazioni

Prerequisiti

- Un database mainframe esistente, ad esempio IBM DB2, IBM Information Management System (IMS) o Virtual Storage Access Method (VSAM), che desideri replicare nel cloud AWS
- Un account AWS attivo
- AWS Direct Connect o AWS Virtual Private Network (AWS VPN) dal tuo ambiente aziendale ad AWS
- Un cloud privato virtuale con una sottorete raggiungibile dalla tua piattaforma legacy

Architecture

Stack tecnologico di origine

Un ambiente mainframe che include almeno uno dei seguenti database:

- Database IBM IMS
- Banca dati IBM DB2
- file VSAM

Stack tecnologico Target

- MSK Amazon
- Amazon Elastic Kubernetes Service (Amazon EKS) e Amazon EKS Anywhere
- Docker
- Un database relazionale AWS o NoSQL come il seguente:
  - Amazon DynamoDB
  - Amazon Relational Database Service (Amazon RDS) per Oracle, Amazon RDS per PostgreSQL o Amazon Aurora
  - Amazon ElastiCache per Redis
  - Amazon Keyspaces (per Apache Cassandra)

Architettura Target

Replica dei dati del mainframe nei database AWS

Il diagramma seguente illustra la replica dei dati mainframe su un database AWS come DynamoDB, Amazon RDS, Amazon o Amazon Keyspaces. ElastiCache La replica avviene quasi in tempo reale utilizzando Precisly Capture and Publisher nell'ambiente mainframe locale, Precisly Dispatcher su Amazon EKS Anywhere nell'ambiente distribuito locale e Precisly Apply Engine e connettori di database nel cloud AWS.

Il diagramma mostra il flusso di lavoro seguente:

1. Precisly Capture ottiene i dati del mainframe dai log del CDC e li conserva in uno storage transitorio interno.
2. Precisly Publisher ascolta le modifiche nella memoria interna dei dati e invia i record CDC a Precisly Dispatcher tramite una connessione. TCP/IP
3. Precisamente Dispatcher riceve i record CDC da Publisher e li invia ad Amazon MSK. Dispatcher crea chiavi Kafka in base alla configurazione dell'utente e a più attività di lavoro per inviare i dati in parallelo. Dispatcher invia una conferma a Publisher quando i record sono stati archiviati in Amazon MSK.
4. Amazon MSK detiene i record CDC nell'ambiente cloud. La dimensione della partizione degli argomenti dipende dai requisiti del sistema di elaborazione delle transazioni (TPS) per la velocità

effettiva. La chiave Kafka è obbligatoria per l'ulteriore trasformazione e l'ordinamento delle transazioni.

5. Il Precisly Apply Engine ascolta i record CDC di Amazon MSK e trasforma i dati (ad esempio, filtrandoli o mappandoli) in base ai requisiti del database di destinazione. È possibile aggiungere logica personalizzata agli script Precisly SQD. (SQD è il linguaggio proprietario di Precisly.) Il Precisly Apply Engine trasforma ogni record CDC in formato Apache Avro o JSON e lo distribuisce su diversi argomenti in base alle esigenze dell'utente.
6. Gli argomenti Kafka di destinazione contengono i record CDC in più argomenti basati sul database di destinazione e Kafka facilita l'ordinamento delle transazioni in base alla chiave Kafka definita. Le chiavi di partizione si allineano con le partizioni corrispondenti per supportare un processo sequenziale.
7. I connettori di database (applicazioni Java personalizzate) ascoltano i record CDC di Amazon MSK e li archiviano nel database di destinazione.
8. Puoi selezionare un database di destinazione in base alle tue esigenze. Questo modello supporta sia i database NoSQL che quelli relazionali.

Ripristino di emergenza

La continuità aziendale è fondamentale per il successo dell'organizzazione. Il cloud AWS offre funzionalità per l'alta disponibilità (HA) e il disaster recovery (DR) e supporta i piani di failover e fallback dell'organizzazione. Questo modello segue una strategia di active/passive DR e fornisce linee guida di alto livello per l'implementazione di una strategia DR che soddisfi i requisiti RTO e RPO.

Il diagramma seguente illustra il flusso di lavoro del DR.

Il diagramma mostra:

1. È necessario un failover semiautomatico in caso di guasto nella regione 1 di AWS. In caso di errore nella Regione 1, il sistema deve avviare le modifiche al routing per connettere Precisly Dispatcher alla Regione 2.
2. Amazon MSK replica i dati tramite mirroring tra regioni. Per questo motivo, durante il failover, il cluster Amazon MSK nella Regione 2 deve essere promosso come leader principale.
3. Il motore di applicazione precisa e i connettori del database sono applicazioni stateless che possono funzionare in qualsiasi regione.

4. La sincronizzazione del database dipende dal database di destinazione. Ad esempio, DynamoDB può utilizzare tabelle globali ElastiCache e datastore globali.

Elaborazione a bassa latenza e ad alto rendimento tramite connettori di database

I connettori di database sono componenti fondamentali in questo modello. I connettori seguono un approccio basato su listener per raccogliere dati da Amazon MSK e inviare transazioni al database tramite elaborazione ad alta velocità e bassa latenza per applicazioni mission-critical (livelli 0 e 1). Il diagramma seguente illustra tale processo.

Questo modello supporta lo sviluppo di un'applicazione personalizzata con utilizzo a thread singolo tramite un motore di elaborazione multithread.

1. Il thread principale del connettore consuma i record CDC da Amazon MSK e li invia al pool di thread per l'elaborazione.
2. I thread del pool di thread elaborano i record CDC e li inviano al database di destinazione.
3. Se tutti i thread sono occupati, i record CDC vengono mantenuti in attesa dalla coda dei thread.
4. Il thread principale attende che tutti i record vengano cancellati dalla coda dei thread e trasferisce gli offset in Amazon MSK.
5. I thread secondari gestiscono gli errori. Se si verificano errori durante l'elaborazione, i messaggi non riusciti vengono inviati all'argomento DLQ (coda di lettere morte).
6. I thread secondari avviano gli aggiornamenti condizionali (vedi Condition expression nella documentazione di DynamoDB), in base al timestamp del mainframe, per evitare duplicazioni o aggiornamenti nel database. out-of-order

Per informazioni su come implementare un'applicazione Kafka consumer con funzionalità multi-threading, consulta il post di blog Multi-Threaded Message Consumption with the Apache Kafka Consumer sul sito Web di Confluent.

Tools (Strumenti)

Servizi AWS

- Amazon Managed Streaming for Apache Kafka (Amazon MSK) è un servizio completamente gestito che ti aiuta a creare ed eseguire applicazioni che utilizzano Apache Kafka per elaborare dati di streaming.

- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire Kubernetes su AWS senza dover installare o mantenere il tuo piano di controllo o i tuoi nodi Kubernetes.
- Amazon EKS Anywhere ti aiuta a distribuire, utilizzare e gestire i cluster Kubernetes eseguiti nei tuoi data center.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS.
- Amazon ti ElastiCache aiuta a configurare, gestire e scalare ambienti di cache in memoria distribuiti nel cloud AWS.
- Amazon Keyspaces (per Apache Cassandra) è un servizio di database gestito che ti aiuta a migrare, eseguire e scalare i carichi di lavoro Cassandra nel cloud AWS.

Altri strumenti

- Precisly Connect integra i dati provenienti da sistemi mainframe legacy come set di dati VSAM o database mainframe IBM in piattaforme cloud e dati di nuova generazione.

Best practice

- Trova la combinazione migliore di partizioni Kafka e connettori multithread per bilanciare prestazioni e costi ottimali. Più istanze Precisly Capture e Dispatcher possono aumentare i costi a causa del maggiore consumo di MIPS (milioni di istruzioni al secondo).
- Evita di aggiungere logica di manipolazione e trasformazione dei dati ai connettori del database. A tale scopo, utilizzate Precisly Apply Engine, che fornisce tempi di elaborazione in microsecondi.
- Crea chiamate periodiche di richiesta o controllo dello stato di salute al database (heartbeats) nei connettori del database per riscaldare frequentemente la connessione e ridurre la latenza.
- Implementa la logica di convalida del pool di thread per comprendere le attività in sospeso nella coda dei thread e attendi che tutti i thread vengano completati prima del prossimo sondaggio di Kafka. Ciò consente di evitare la perdita di dati in caso di arresto anomalo di un nodo, contenitore o processo.
- Esponi le metriche di latenza attraverso gli endpoint sanitari per migliorare le capacità di osservabilità tramite dashboard e meccanismi di tracciamento.

Epiche

Prepara l'ambiente di origine (locale)

Operazione	Description	Competenze richieste
Configura il processo mainframe (batch o utilità online) per avviare il processo CDC dai database mainframe.	1. Identifica l'ambiente mainframe. 2. Identifica i database mainframe che saranno coinvolti nel processo CDC. 3. Nell'ambiente mainframe , sviluppate un processo che avvii lo strumento CDC per acquisire le modifiche nel database di origine. Per istruzioni, consultat e la documentazione del mainframe. 4. Documenta il processo CDC, inclusa la configura zione. 5. Implementa il processo sia in ambienti di test che di produzione.	Ingegnere del mainframe
Attiva i flussi di log del database mainframe.	1. Configura i flussi di log nell'ambiente mainframe per acquisire i log CDC. Per istruzioni, consultat e la documentazione del mainframe. 2. Verifica i flussi di log per assicurarti che acquisiscano i dati necessari.	Specialista in database mainframe

Operazione	Description	Competenze richieste
	3. Implementa i flussi di log in ambienti di test e produzione.	
Utilizzate il componente Capture per acquisire i record CDC.	1. Installa e configura il componente Precisly Capture nell'ambiente mainframe. Per istruzioni, consultate la documentazione di Precisly. 2. Verificate la configurazione per assicurarvi che il componente Capture funzioni correttamente. 3. Imposta un processo di replica per replicare i record CDC acquisiti tramite il componente Capture. 4. Documenta la configurazione di Capture per ogni database di origine. 5. Sviluppa un sistema di monitoraggio per garantire che il componente Capture raccolga i log correttamente nel tempo. 6. Implementa l'installazione e le configurazioni negli ambienti di test e produzione.	Ingegnere mainframe, Precisly Connect SME

Operazione	Description	Competenze richieste
Configura il componente Publisher per ascoltare il componente Capture.	1. Installa e configura il componente Precisly Publisher nell'ambiente mainframe. Per istruzioni, consultate la documentazione di Precisly. 2. Verificate la configurazione per assicurarvi che il componente Publisher funzioni correttamente. 3. Impostate un processo di replica per pubblicare i record CDC nel componente Precisly Dispatcher di Publisher. 4. Documenta la configurazione di Publisher. 5. Sviluppa un sistema di monitoraggio per garantire che il componente Publisher funzioni correttamente nel tempo. 6. Implementa l'installazione e le configurazioni negli ambienti di test e di produzione.	Ingegnere mainframe, Precisly Connect SME

Operazione	Description	Competenze richieste
Esegui il provisioning di Amazon EKS Anywhere nell'ambiente distribuito locale.	1. Installa Amazon EKS Anywhere sull'infrastruttura locale e assicurati che sia configurata correttamente. Per istruzioni, consulta la documentazione di Amazon EKS Anywhere. 2. Configura un ambiente di rete sicuro per il cluster Kubernetes, inclusi i firewall. 3. Implementa e testa la distribuzione di applicazioni di esempio nel cluster Amazon EKS Anywhere. 4. Implementa funzionalità di scalabilità automatica per il cluster. 5. Sviluppa e implementa procedure di backup e disaster recovery.	DevOps ingegnere

Operazione	Description	Competenze richieste
Distribuisci e configura il componente Dispatcher nell'ambiente distribuito per pubblicare gli argomenti nel cloud AWS.	1. Configura e containerizza il componente Precisly Dispatcher. Per istruzioni, consulta la documentazione di Precisly. 2. Implementa l'immagine Dispatcher Docker nell'ambiente locale Amazon EKS Anywhere. 3. Configura una connessione sicura tra il cloud AWS e Dispatcher. 4. Sviluppa un sistema di monitoraggio per garantire che il componente Dispatcher funzioni correttamente nel tempo. 5. Implementa l'installazione e le configurazioni negli ambienti di test e produzione.	DevOps ingegnere, Precisly Connect SME

Preparare l'ambiente di destinazione (AWS)

Operazione	Description	Competenze richieste
Effettua il provisioning di un cluster Amazon EKS nella regione AWS designata.	1. Accedi al tuo account AWS e configuralo per assicurarti che siano disponibili le autorizzazioni necessarie per creare e gestire il cluster Amazon EKS.	DevOps ingegnere, amministratore di rete

Operazione	Description	Competenze richieste
	2. Crea un cloud privato virtuale (VPC) e sottoreti nella regione AWS selezionata. Per istruzioni, consulta la documentazione di Amazon EKS. 3. Crea e configura i gruppi di sicurezza di rete necessari per consentire le comunicazioni tra il cluster Amazon EKS e altre risorse nel VPC. Per ulteriori informazioni, consulta la documentazione di Amazon EKS. 4. Crea il cluster Amazon EKS e configuralo con la dimensione del gruppo di nodi e i tipi di istanza corretti. 5. Convalida il cluster Amazon EKS distribuendo un'applicazione di esempio.	

Operazione	Description	Competenze richieste
Esegui il provisioning di un cluster MSK e configura gli argomenti Kafka applicabili.	1. Configura il tuo account AWS per assicurarti che siano disponibili le autorizzazioni necessarie per creare e gestire il cluster MSK. 2. Crea e configura i gruppi di sicurezza di rete necessari per consentire le comunicazioni tra il cluster MSK e altre risorse nel VPC. Per ulteriori informazioni, consulta la documentazione di Amazon VPC. 3. Crea il cluster MSK e configuralo per includere gli argomenti di Kafka che verranno utilizzati dall'applicazione. Per ulteriori informazioni, consulta la documentazione di Amazon MSK.	DevOps ingegnere, amministratore di rete

Operazione	Description	Competenze richieste
Configura il componente Apply Engine per ascoltare gli argomenti di Kafka replicati.	1. Configura e containerizza il componente Precisly Apply Engine. 2. Distribuisci l'immagine Docker di Apply Engine nel cluster Amazon EKS del tuo account AWS. 3. Configura Apply Engine per ascoltare gli argomenti di MSK. 4. Sviluppa e configura uno script SQD in Apply Engine per gestire il filtraggio e la trasformazione. Per ulteriori informazioni, consulta la documentazione di Precisly. 5. Implementa Apply Engine in ambienti di test e produzione.	Precisamente Connect SME

Operazione	Description	Competenze richieste
Effettua il provisioning di istanze DB nel cloud AWS.	1. Configura il tuo account AWS per assicurarti che siano disponibili le autorizzazioni necessarie per creare e gestire cluster e tabelle DB. Per istruzioni, consulta la documentazione AWS per il servizio di database AWS che desideri utilizzare. (Per i collegamenti, consulta la sezione Risorse). 2. Crea un VPC e delle sottoreti nella regione AWS selezionata. 3. Crea e configura i gruppi di sicurezza di rete necessari per consentire le comunicazioni tra le istanze DB e altre risorse nel VPC. 4. Crea i database e configurali per includere le tabelle che l'applicazione utilizzerà. 5. Progetta e convalida gli schemi del database.	Ingegnere dei dati, ingegnere DevOps

Operazione	Description	Competenze richieste
Configura e distribuisci connettori di database per ascoltare gli argomenti pubblicati da Apply Engine.	1. Progetta connettori di database per connetter e gli argomenti di Kafka con i database AWS che hai creato nei passaggi precedenti. 2. Sviluppa i connettori in base al database di destinazione. 3. Configura i connettori per ascoltare gli argomenti di Kafka pubblicati da Apply Engine. 4. Implementa i connettori nel cluster Amazon EKS.	Sviluppatore di app, architetto del cloud, ingegnere dei dati

Configura la continuità aziendale e il disaster recovery

Operazione	Description	Competenze richieste
Definisci gli obiettivi di disaster recovery per le tue applicazioni aziendali.	1. Definisci gli obiettivi RPO e RTO per le pipeline CDC in base alle esigenze aziendali e all'analisi dell'impatto. 2. Definisci le procedure di comunicazione e notifica per garantire che tutte le parti interessate siano a conoscenza del piano di disaster recovery. 3. Determina il budget e le risorse necessari per implementare il piano di disaster recovery.	Architetto del cloud, ingegnere dei dati, proprietario dell'app

Operazione	Description	Competenze richieste
	4. Documenta gli obiettivi di disaster recovery, inclusi gli obiettivi RPO e RTO.	
Progetta strategie di disaster recovery basate su RTO/RPO definiti.	1. Determina le strategie di disaster recovery più appropriate per le pipeline CDC in base ai tuoi requisiti di criticità e ripristino. 2. Definisci l'architettura e la topologia del disaster recovery. 3. Definisci le procedure di failover e failback per le pipeline CDC per garantire che possano essere trasferite rapidamente e senza problemi alla regione di backup. 4. Documenta le strategie e le procedure di disaster recovery e assicurati che tutte le parti interessa te abbiano una chiara comprensione del progetto.	Architetto del cloud, ingegnere dei dati

Operazione	Description	Competenze richieste
Fornisci cluster e configurazioni di disaster recovery.	1. Esegui il provisioning di una regione AWS secondaria per il disaster recovery. 2. Nella regione AWS secondaria, crea un ambiente identico alla regione AWS principale. 3. Configura Apache Kafka MirrorMaker tra la regione principale e quella secondaria. Per ulteriori informazioni, consulta la documentazione di Amazon MSK. 4. Configura le applicazioni in standby nella regione secondaria. 5. Configura le repliche del database tra le regioni primarie e secondarie.	DevOps ingegnere, amministratore di rete, architetto cloud

Operazione	Description	Competenze richieste
Testa la pipeline CDC per il disaster recovery.	1. Definisci l'ambito e gli obiettivi del test di disaster recovery per la pipeline CDC, inclusi gli scenari di test e l'RTO da raggiunge re. 2. Identifica l'ambiente e l'infrastruttura di test per condurre il test di disaster recovery. 3. Prepara i set di dati e lo script di test per simulare scenari di errore. 4. Verifica l'integrità e la coerenza dei dati per garantire che non vi siano perdite di dati.	Proprietario dell'app, ingegnere dei dati, architetto cloud

Risorse correlate

Risorse AWS

- Amazon DynamoDB
- Espressioni di condizione con Amazon DynamoDB
- Amazon EKS
- Amazon EKS Anywhere
- Amazon ElasticCache
- Amazon Keyspaces
- Amazon MSK
- Amazon RDS e Amazon Aurora
- Amazon VPC

Risorse Precisly Connect

- Panoramica di Precisly Connect
- Modifica l'acquisizione dei dati con Precisly Connect

Risorse confluenti

- Consumo di messaggi multithread con Apache Kafka Consumer

Pianifica i lavori per Amazon RDS for PostgreSQL e Aurora PostgreSQL utilizzando Lambda e Secrets Manager

Yaser Raja, Amazon Web Services

Riepilogo

Per i database e i database locali ospitati su istanze Amazon Elastic Compute Cloud EC2 (Amazon), gli amministratori di database utilizzano spesso l'utilità cron per pianificare i lavori.

Ad esempio, un lavoro per l'estrazione dei dati o un lavoro per l'eliminazione dei dati può essere facilmente pianificato utilizzando cron. Per questi lavori, le credenziali del database sono in genere codificate o archiviate in un file di proprietà. Tuttavia, quando esegui la migrazione ad Amazon Relational Database Service (Amazon RDS) o Amazon Aurora PostgreSQL Compatible Edition, perdi la possibilità di accedere all'istanza host per pianificare cron job.

Questo modello descrive come utilizzare AWS Lambda e pianificare i lavori per Gestione dei segreti AWS i database compatibili con Amazon RDS for PostgreSQL e Aurora PostgreSQL dopo la migrazione.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un database compatibile con Amazon RDS per PostgreSQL o Aurora PostgreSQL

Limitazioni

- Un processo deve essere completato entro 15 minuti, che è il limite di timeout della funzione Lambda. Per altri limiti, consulta la AWS Lambda documentazione.
- Il codice Job deve essere scritto in un linguaggio supportato da Lambda.

Architecture

Stack tecnologico di origine

Questo stack include lavori scritti in linguaggi come Bash, Python e Java. Le credenziali del database sono memorizzate nel file delle proprietà e il lavoro viene pianificato utilizzando Linux cron.

Stack tecnologico Target

Questo stack ha una funzione Lambda che utilizza le credenziali archiviate in Secrets Manager per connettersi al database ed eseguire l'attività. La funzione Lambda viene avviata a intervalli pianificati utilizzando Amazon Events. CloudWatch

Architettura Target

Tools (Strumenti)

- Amazon CloudWatch Events offre un flusso quasi in tempo reale di eventi di sistema che descrivono i cambiamenti nelle AWS risorse. Utilizzando semplici regole che puoi configurare rapidamente, puoi abbinare gli eventi e indirizzarli a una o più funzioni o flussi di destinazione. CloudWatch Gli eventi vengono a conoscenza dei cambiamenti operativi man mano che si verificano. Risponde a questi cambiamenti operativi e adotta le azioni correttive necessarie, inviando messaggi per rispondere all'ambiente, attivando funzioni, apportando modifiche e acquisendo informazioni sullo stato. Puoi anche utilizzare CloudWatch Events per pianificare azioni automatiche che si avviano automaticamente in determinati momenti utilizzando le espressioni cron o rate.
- AWS Lambda è un servizio di elaborazione che consente di eseguire del codice senza la necessità di effettuare il provisioning o la gestione dei server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo. Paghi solo per il tempo di elaborazione che consumi; non è previsto alcun addebito quando il codice non è in esecuzione. Con Lambda, puoi eseguire codice praticamente per qualsiasi tipo di applicazione o servizio di backend senza alcuna amministrazione. Lambda esegue il codice su un'infrastruttura di elaborazione ad alta disponibilità e gestisce tutte le risorse di calcolo, tra cui la manutenzione di server e sistemi operativi, il provisioning della capacità e il ridimensionamento automatico, il monitoraggio del codice e la registrazione. Tutto ciò che devi fare è fornire il codice in una delle lingue supportate da Lambda.
- Gestione dei segreti AWSti aiuta a proteggere i segreti per l'accesso alle tue applicazioni, servizi e risorse IT. Puoi ruotare, gestire e recuperare facilmente le credenziali del database, le chiavi API e altri segreti durante tutto il loro ciclo di vita. Gli utenti e le applicazioni recuperano i segreti chiamando Secrets Manager APIs, che elimina la necessità di codificare le informazioni sensibili in testo normale. Secrets Manager offre una rotazione segreta con integrazione integrata per Amazon RDS, Amazon Redshift e Amazon DocumentDB. Il servizio è estensibile ad altri tipi di segreti, tra cui chiavi e token API. OAuth Secrets Manager consente di controllare l'accesso ai segreti

utilizzando autorizzazioni granulari e di controllare centralmente la rotazione segreta per le risorse nei servizi di terze parti e in locale. Cloud AWS

Epiche

Memorizza le credenziali del database in Secrets Manager

Operazione	Description	Competenze richieste
Crea un utente del database per la funzione Lambda.	È buona norma utilizzare utenti di database separati per diverse parti dell'appl icazione. Se esiste già un utente del database separato per i tuoi cron job, usalo. Altrimenti, crea un nuovo utente del database. Per ulteriori informazioni, consulta Gestione degli utenti e dei ruoli PostgreSQL (post del blog).AWS	DBA
Memorizza le credenziali del database come segreto in Secrets Manager.	Segui le istruzioni riportate in Creare un database segreto (documentazione di Secrets Manager).	DBA, DevOps

Crea il codice per la funzione Lambda

Operazione	Description	Competenze richieste
Scegli un linguaggio di programmazione supportato da Lambda.	Per un elenco delle lingue supportate, consulta Lambda runtimes (documentazione Lambda).	Developer

Operazione	Description	Competenze richieste
Scrivi la logica per recuperare le credenziali del database da Secrets Manager.	Per un codice di esempio, vedi Come fornire in modo sicuro le credenziali del database alle funzioni Lambda utilizzando Gestione dei segreti AWS (AWS post del blog).	Developer
Scrivi la logica per eseguire l'attività pianificata del database.	Esegui la migrazione del codice esistente per il processo di pianificazione che stai utilizzando in locale alla funzione Lambda. Per ulteriori informazioni, consulta Implementazione delle funzioni Lambda (documentazione Lambda).	Developer

Distribuisci il codice e crea la funzione Lambda

Operazione	Description	Competenze richieste
Crea il pacchetto di distribuzione della funzione Lambda.	Questo pacchetto contiene il codice e le sue dipendenze. Per ulteriori informazioni, consulta Pacchetti di distribuzione (documentazione Lambda).	Developer
Creazione della funzione Lambda	Nella console Lambda, scegli Crea funzione, inserisci il nome di una funzione, scegli l'ambiente di runtime, quindi scegli Crea funzione.	DevOps

Operazione	Description	Competenze richieste
Carica il pacchetto di distribuzione.	Scegli la funzione Lambda che hai creato per aprirne la configurazione. Puoi scrivere il codice direttamente nella sezione codice o caricare il pacchetto di distribuzione. Per caricare il pacchetto, vai alla sezione Codice funzione, scegli il tipo di immissione del codice per caricare un file.zip, quindi seleziona il pacchetto.	DevOps
Configura la funzione Lambda in base alle tue esigenze.	Ad esempio, puoi impostare il parametro Timeout sulla durata prevista per la funzione Lambda. Per ulteriori informazioni, consulta Configurazione delle opzioni delle funzioni (documentazione Lambda).	DevOps
Imposta le autorizzazioni per il ruolo della funzione Lambda per accedere a Secrets Manager.	Per istruzioni, consulta Utilizzare i segreti nelle AWS Lambda funzioni (documentazione di Secrets Manager).	DevOps
Prova la funzione Lambda.	Avvia manualmente la funzione Lambda per assicurarti che funzioni come previsto.	DevOps

Pianifica la funzione Lambda utilizzando Events CloudWatch

Operazione	Description	Competenze richieste
Crea una regola per eseguire la funzione Lambda secondo una pianificazione.	Pianifica la funzione Lambda utilizzando CloudWatch Events. Per istruzioni, consulta Pianifica le funzioni Lambda utilizzando CloudWatch gli eventi (tutorial sugli CloudWatch eventi).	DevOps

Risorse correlate

- Gestione dei segreti AWS
- Guida introduttiva a Lambda
- Creazione di una regola per CloudWatch gli eventi che si attiva su un evento
- AWS Lambda Limiti
- Interroga il AWS database dalla tua applicazione serverless (post sul blog)

Invia notifiche per un'istanza di database Amazon RDS for SQL Server utilizzando un server SMTP locale e Database Mail

Nishad Mankar, Amazon Web Services

Riepilogo

Database Mail (documentazione Microsoft) invia messaggi di posta elettronica, come notifiche o avvisi, da un database di Microsoft SQL Server utilizzando un server SMTP (Simple Mail Transfer Protocol). La documentazione di Amazon Relational Database Service (Amazon RDS) per Microsoft SQL Server fornisce istruzioni per utilizzare Amazon Simple Email Service (Amazon SES) come server SMTP per Database Mail. Per ulteriori informazioni, consulta Utilizzo di Database Mail in Amazon RDS for SQL Server. Come configurazione alternativa, questo modello spiega come configurare Database Mail per inviare e-mail da un'istanza di database Amazon RDS for SQL Server (DB) utilizzando un server SMTP locale come server di posta.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un'istanza database Amazon RDS che esegue un'edizione Standard o Enterprise di SQL Server
- L'indirizzo IP o il nome host del server SMTP locale
- Una regola del gruppo di sicurezza in entrata che consente le connessioni all'istanza DB di Amazon RDS for SQL Server dall'indirizzo IP del server SMTP
- Una connessione, ad esempio una connessione AWS Direct Connect, tra la rete locale e il cloud privato virtuale (VPC) che contiene l'istanza database Amazon RDS

Limitazioni

- Le edizioni Express di SQL Server non sono supportate.
- Per ulteriori informazioni sulle limitazioni, consulta Limitazioni nell'uso di Database Mail su Amazon RDS for SQL Server nella documentazione di Amazon RDS.

Versioni del prodotto

- Edizioni Standard ed Enterprise delle versioni di SQL Server supportate in RDS

Architecture

Stack tecnologico Target

- Istanza di database Amazon RDS per SQL Server
- Regola di inoltro di Amazon Route 53
- Posta elettronica database
- Server SMTP locale
- Microsoft SQL Server Management Studio (SSMS)

Architettura Target

L'immagine seguente mostra l'architettura di destinazione per questo modello. Quando si verifica un evento o un'azione che avvia una notifica o un avviso relativo all'istanza del database, Amazon RDS for SQL Server utilizza Database Mail per inviare una notifica e-mail. Database Mail utilizza il server SMTP locale per inviare l'e-mail.

Tools (Strumenti)

Servizi AWS

- Amazon Relational Database Service (Amazon RDS) per Microsoft SQL Server ti aiuta a configurare, gestire e scalare un database relazionale SQL Server nel cloud AWS.
- Amazon Route 53 è un servizio Web DNS altamente scalabile e disponibile.

Altri strumenti

- Database Mail è uno strumento che invia messaggi di posta elettronica, come notifiche e avvisi, dal motore di database di SQL Server agli utenti.
- Microsoft SQL Server Management Studio (SSMS) è uno strumento per la gestione di SQL Server, che include l'accesso, la configurazione e l'amministrazione dei componenti di SQL Server.
  In questo modello, usi SSMS per eseguire i comandi SQL per configurare Database Mail su un'istanza DB di Amazon RDS for SQL Server.

Epiche

Abilita la connettività di rete con il server SMTP locale

Operazione	Description	Competenze richieste
Rimuovi Multi-AZ dall'istanza DB RDS.	Se utilizzi un'istanza DB RDS multi-zona, converti l'istanza Multi-AZ in un'istanz a Single-AZ. Al termine della configurazione di Database Mail, riconvertirai l'istanza DB in una distribuzione Multi-AZ. La configurazione di Database Mail funziona quindi sia nel nodo primario che in quello secondario. Per istruzioni, vedere Rimozione di Multi-AZ da un'istanza DB di Microsoft SQL Server.	DBA
Crea un elenco di indirizzi consentiti per l'endpoint o l'indirizzo IP di Amazon RDS sul server SMTP locale.	Il server SMTP è esterno alla rete AWS. Sul server SMTP locale, crea un elenco di autorizzazioni che consenta al server di comunicare con l'endpoint o l'indirizzo IP in uscita per l'istanza Amazon RDS o l'istanza Amazon Elastic Compute Cloud (Amazon EC2) ospitata su Amazon RDS. Questa procedura varia da organizza zione a organizzazione. Per ulteriori informazioni sull'endp oint dell'istanza DB, vedere Individuazione dell'endpoint	DBA

Operazione	Description	Competenze richieste
	e del numero di porta dell'istanza DB.	
Rimuovi le restrizioni sulla porta 25.	Per impostazione predefinita, AWS limita la porta 25 sulle EC2 istanze. Per rimuovere la restrizione sulla porta 25, procedi come segue: 1. Accedi con il tuo account AWS, quindi apri il modulo Richiesta di rimozione delle limitazioni all'invio di e-mail. 2. Inserisci il tuo indirizzo e-mail in modo che AWS Support possa contattarti con aggiornamenti sulla tua richiesta. 3. Fornisci le informazioni richieste nel campo Descrizione del caso d'uso. 4. Seleziona Invia. Note • Se hai istanze in più di una regione AWS, invia una richiesta separata per ogni regione. • L'elaborazione della richiesta può richiedere fino a 48 ore.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Aggiungi una regola Route 53 per risolvere le query DNS per il server SMTP.	Usa Route 53 per risolvere le query DNS tra le tue risorse AWS e il server SMTP locale. È necessario creare una regola che inoltri le query DNS al dominio del server SMTP, ad esempio. `example.com` Per istruzioni, consulta Creazione di regole di inoltro nella documentazione di Route 53.	Amministratore di rete

Configurare Database Mail sull'istanza DB di Amazon RDS for SQL Server

Operazione	Description	Competenze richieste
Abilita Database Mail.	Crea un gruppo di parametri per Database Mail, imposta il `database mail xps` parametro su e quindi associa 1 il gruppo di parametri Database Mail all'istanza DB RDS di destinazione. Per istruzioni, consulta Enabling Database Mail nella documentazione di Amazon RDS. Non procedere alla sezione Configurazione di Database Mail in queste istruzioni. La configurazione del server SMTP locale è diversa da quella di Amazon SES.	DBA

Operazione	Description	Competenze richieste
Effettua la connessione all'istanza database.	Da un host bastion, usa Microsoft SQL Server Management Studio (SSMS) per connetterti all'istanza del database Amazon RDS for SQL Server. Per istruzioni, vedere Connessione a un'istanza DB che esegue il motore di database Microsoft SQL Server. In caso di errori, consulta i riferimenti per la risoluzione dei problemi di connessione nella sezione Risorse correlate.	DBA

Operazione	Description	Competenze richieste
Crea il profilo.	In SSMS, inserisci la seguente istruzione SQL per creare il profilo Database Mail. Sostituisci i valori seguenti: • Per `profile_name`, inserisci un nome per il nuovo profilo. • Per `description`, inserisci una breve descrizione del nuovo profilo. Per ulteriori informazioni su questa stored procedure e sui relativi argomenti, vedere sysmail_add_profile_sp nella documentazione Microsoft. `EXECUTE msdb.dbo.sysmail_add_profile_sp @profile_name = 'SQL Alerts profile', @description = 'Profile used for sending outgoing notifications using OM SMTP Server.';`	DBA

Operazione	Description	Competenze richieste
Aggiungi i principali al profilo.	Immettere la seguente istruzione SQL per aggiungere i principali pubblici o privati al profilo Database Mail. Un principal è un'entità che può richiedere risorse di SQL Server. Sostituisci i valori seguenti: • Per `profile_name`, inserisci il nome del profilo che hai creato in precedenza. • Per `principal_name`, inserisci il nome dell'utente o del ruolo del database. Questo valore deve essere mappato a un utente di autenticazione di SQL Server, a un utente di autenticazione di Windows o a un gruppo di autenticazione di Windows. Per ulteriori informazioni su questa stored procedure e sui relativi argomenti, vedere sysmail_add_principalprofile_sp nella documentazione Microsoft. `EXECUTE msdb.dbo.sysmail_add_principalprofile_sp`	DBA

Operazione	Description	Competenze richieste
	`@profile_name = 'SQL Alerts profile', @principal_name = 'public', @is_default = 1 ;`	

Operazione	Description	Competenze richieste
Crea l'account.	Immettere la seguente istruzione SQL per creare l'account Database Mail. Sostituisci i valori seguenti: • Per `account_name`, inserisci un nome per il nuovo account. • Per `description`, inserisci una breve descrizione del nuovo account. • Per `email_address`, inserisci l'indirizzo e-mail da cui inviare i messaggi di Database Mail. • Per `display_address`, inserisci un nome visualizzato da utilizzare per i messaggi in uscita per questo account, ad esempio `SQL Server Automated Notification`. Puoi anche usare il valore per `email_address` cui hai inserito. • Per `mailserver_name`, inserisci il nome o l'indirizzo IP del server di posta SMTP. • Per `port`, lascia il valore di. `25` • Per `enable_ssl`, lascia il valore `1` o inserisci `0` se	DBA

Operazione	Description	Competenze richieste
	non vuoi che Database Mail crittografi le comunicazioni utilizzando SSL. • Per `username`, inserisci il nome utente per accedere al server di posta SMTP. Se il server non richiede l'autenticazione, immettere. `NULL` • Per `password`, inserire la password per accedere al server di posta SMTP. Se il server non richiede l'autenti cazione, immettere. `NULL` Per ulteriori informazioni su questa stored procedure e sui relativi argomenti, vedere sysmail_add_account_sp nella documentazione Microsoft. `EXECUTE msdb.dbo.sysmail_add_accoun t_sp @account_name = 'SQL Alerts account', @description = 'Database Mail account for sending outgoing notifications.', @email_address = 'xyz@example.com', @display_name = 'xyz@example.com', @mailserver_name = 'test_smtp.example .com',`	

Operazione	Description	Competenze richieste

	`@port = 25,`
	`@enable_ssl = 1,`
	`@username = 'SMTP-username',`
	`@password = 'SMTP-password';`	

Operazione	Description	Competenze richieste
Aggiungi l'account al profilo.	Immettere la seguente istruzione SQL per aggiunger e l'account Database Mail al profilo Database Mail. Sostituisci i valori seguenti: • Per `profile_name` , inserisci il nome del profilo che hai creato in precedenz a. • Per `account_name` , inserisci il nome dell'account che hai creato in precedenz a. Per ulteriori informazioni su questa stored procedure e sui relativi argomenti, vedere sysmail_add_profileaccount_ sp nella documentazione Microsoft. `EXECUTE msdb.dbo. sysmail_add_profil eaccount_sp @profile_name = 'SQL Alerts profile', @account_name = 'SQL Alerts account', @sequence_number = 1;`	DBA

Operazione	Description	Competenze richieste
(Facoltativo) Aggiungi Multi-AZ all'istanza DB RDS.	Se desideri aggiungere Multi-AZ con Database Mirroring (DBM) o Always On Availability Groups (AGs), consulta le istruzioni in Aggiungere Multi-AZ a un'istanza DB di Microsoft SQL Server.	DBA

Risorse correlate

- Utilizzo di Database Mail su Amazon RDS per SQL Server (documentazione Amazon RDS)
- Utilizzo dei file allegati (documentazione Amazon RDS)
- Risoluzione dei problemi di connessione all'istanza DB di SQL Server (documentazione Amazon RDS)
- Impossibile connettersi all'istanza database Amazon RDS (documentazione Amazon RDS)

Configura il disaster recovery per SAP su IBM Db2 su AWS

Ambarish Satarkar e Debasis Sahoo, Amazon Web Services

Riepilogo

Questo modello delinea i passaggi per configurare un sistema di disaster recovery (DR) per carichi di lavoro SAP con IBM Db2 come piattaforma di database, in esecuzione sul cloud Amazon Web Services (AWS). L'obiettivo è fornire una soluzione a basso costo per garantire la continuità aziendale in caso di interruzione.

Il modello utilizza l'approccio della luce pilota. Implementando il DR pilota light su AWS, puoi ridurre i tempi di inattività e mantenere la continuità aziendale. L'approccio pilota si concentra sulla configurazione di un ambiente DR minimo in AWS, che include un sistema SAP e un database Db2 in standby, sincronizzato con l'ambiente di produzione.

Questa soluzione è scalabile. Se necessario, è possibile estenderla a un ambiente di disaster recovery completo.

Prerequisiti e limitazioni

Prerequisiti

- Un'istanza SAP in esecuzione su un'istanza Amazon Elastic Compute Cloud (Amazon EC2)
- Un database IBM Db2
- Un sistema operativo supportato da SAP Product Availability Matrix (PAM)
- Nomi host di database fisici diversi per gli host di database di produzione e di standby
- Un bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) in ogni regione AWS con Replicazione multiregione (CRR) abilitata

Versioni del prodotto

- Database IBM Db2 versione 11.5.7 o successiva

Architecture

Stack tecnologico Target

- Amazon EC2
- Amazon Simple Storage Service (Amazon S3)
- Amazon Virtual Private Cloud (peering VPC)
- Amazon Route 53
- IBM Db2 High Availability Disaster Recovery (HADR)

Architettura di destinazione

Questa architettura implementa una soluzione DR per carichi di lavoro SAP con Db2 come piattaforma di database. Il database di produzione viene distribuito nella regione AWS 1 e un database di standby viene distribuito in una seconda regione. Il database di standby è denominato sistema DR. Il database Db2 supporta più database in standby (fino a tre). Utilizza Db2 HADR per configurare il database DR e automatizzare la spedizione dei log tra i database di produzione e quelli di standby.

In caso di emergenza che renda indisponibile la Regione 1, il database di standby nella regione DR assume il ruolo di database di produzione. Gli application server SAP possono essere creati in anticipo o utilizzando AWS Elastic Disaster Recovery o Amazon Machine Image (AMI) per soddisfare i requisiti RTO (Recovery Time Objective). Questo modello utilizza un AMI.

Db2 HADR implementa una configurazione di produzione in standby, in cui la produzione funge da server principale e tutti gli utenti sono collegati ad essa. Tutte le transazioni vengono scritte in file di registro, che vengono trasferiti al server di standby tramite TCP/IP. Il server di standby aggiorna il database locale trasferendo i record di registro trasferiti, il che aiuta a garantire che siano mantenuti sincronizzati con il server di produzione.

Il peering VPC viene utilizzato in modo che le istanze nella regione di produzione e nella regione DR possano comunicare tra loro. Amazon Route 53 indirizza gli utenti finali verso le applicazioni Internet.

1. Crea un'AMI del server delle applicazioni nella regione 1 e copia l'AMI nella regione 2. Utilizza l'AMI per avviare i server nella Regione 2 in caso di emergenza.
2. Imposta la replica Db2 HADR tra il database di produzione (nella Regione 1) e il database di standby (nella Regione 2).
3. Modifica il tipo di EC2 istanza in modo che corrisponda all'istanza di produzione in caso di emergenza.

4. Nella Regione 1, `LOGARCHMETH1` è impostato su`db2remote: S3 path`.
5. Nella Regione 2, `LOGARCHMETH1` è impostato su`db2remote: S3 path`.
6. La replica tra regioni viene eseguita tra i bucket S3.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon Route 53 è un servizio Web DNS altamente scalabile e disponibile.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito. Questa rete virtuale è simile a una rete tradizionale che gestiresti nel tuo data center, con i vantaggi dell'utilizzo dell'infrastruttura scalabile di AWS. Questo modello utilizza il peering VPC.

Best practice

- La rete svolge un ruolo chiave nel decidere la modalità di replica HADR. Per il DR in tutte le regioni AWS, ti consigliamo di utilizzare la modalità Db2 HADR ASYNC o SUPERASYNC.
- Per ulteriori informazioni sulle modalità di replica per Db2 HADR, consulta la documentazione IBM.
- Puoi utilizzare la Console di gestione AWS o l'AWS Command Line Interface (AWS CLI) per creare una nuova AMI del tuo sistema SAP esistente. È quindi possibile utilizzare l'AMI per ripristinare il sistema SAP esistente o per creare un clone.
- AWS Systems Manager Automation può aiutarti con le attività comuni di manutenzione e distribuzione di EC2 istanze e altre risorse AWS.
- AWS offre diversi servizi nativi per monitorare e gestire l'infrastruttura e le applicazioni su AWS. Servizi come Amazon CloudWatch e AWS CloudTrail possono essere utilizzati rispettivamente per monitorare l'infrastruttura sottostante e le operazioni API. Per ulteriori dettagli, consulta SAP on AWS — IBM Db2 HADR with Pacemaker.

Epiche

Prepara l'ambiente

Operazione	Description	Competenze richieste
Controlla il sistema e i registri.	1. Verificare che il sistema SAP di produzione su Db2 sia configurato. 2. Verifica che il backup dei log sia attivato e configurato per salvare i log nel bucket S3. Questo può essere verificato tramite il parametro Db2. `LOGARCHMETH1` 3. Crea un'AMI dell'application server aggiuntivo.	Amministratore AWS, amministratore SAP Basis

Configura i server e la replica

Operazione	Description	Competenze richieste
Crea i server SAP e di database.	1. Per distribuire l'infrastruttura per la regione DR, utilizza uno CloudFormation script AWS o un AMI dell'ista nza di produzione. Come parte dell'approccio pilot light, puoi utilizzare un' EC2 istanza più piccola della stessa famiglia dell'ista nza di produzione. Ad esempio, se il tipo di istanza di produzione è `r6i.12xla rge` , puoi utilizzare il tipo	Amministratore SAP Basis

Operazione	Description	Competenze richieste
	di `r6i.xlarge` istanza per la build DR. Tuttavia, assicurati di allocare la stessa capacità di storage sull'istanza DR per ripristin are il backup del database di produzione. 2. Crea punti di montaggio Amazon Elastic File System (Amazon EFS) per `/sapmnt/<SID>/` e assicurati che sia impostato per essere replicato dal sistema primario. 3. Esegui un backup COMPLETO del database (online o offline) dal sistema di produzione. Utilizzerai questo backup per creare il database DR. 4. Nel sistema DR, utilizza il metodo di copia del sistema SAP Software Provision ing Manager (SWPM) con Using system copy with backup/restore per creare il HA/DR sistema DR SAP. 5. Quando richiesto da SWPM, ripristinate il database in DR con il backup che avete prelevato dalla produzione. Il database DR sarà in attesa di rollforward.	

Operazione	Description	Competenze richieste
	Lo stato di rollforward pending viene impostato di default dopo il ripristino del backup completo. Lo stato di rollforwa rd pending indica che il database è in fase di ripristin o e che potrebbe essere necessario applicare alcune modifiche. Per ulteriori informazioni, consulta la documentazione IBM.	

Operazione	Description	Competenze richieste
Controlla la configurazione.	1. Per configurare l'archivi azione dei log per HADR, sia i database di produzion e che quelli DR devono essere in grado di recuperar e automaticamente i log da tutte le posizioni di archiviaz ione dei log. Verificare che il `LOGARCHMETH1` parametro nel database DR sia impostato sulla stessa posizione del database di produzione. Se la stessa posizione non è accessibi le a causa di limitazioni regionali, assicuratevi che il sistema DR possa recuperare automatic amente i registri dal sistema primario. 2. Per abilitare TCP/IP le porte per l'abilitazione della replica del database, modificate gli host `/etc/services` di produzion e e DR aggiungendo le due voci seguenti. Nel codice, `<SID>` fa riferimen to all'ID di sistema (SID) del database Db2 (ad esempio,). PR1 `<SID>_HADR_1 55001/tcp # DB2 HADR Port1`	Amministratore AWS, amministratore SAP Basis

Operazione	Description	Competenze richieste
	<SID>_HADR_2 55002/tcp # DB2 HADR Port2 Verifica che entrambe le porte consentano il traffico in entrata e in uscita tra la porta principale e quella di standby. 3. /etc/hosts Effettua il check-in negli host di produzione e DR per verificare che i nomi host degli host di produzione e di standby puntino agli indirizzi IP corretti.	

Operazione	Description	Competenze richieste

Imposta la replica dal DB di produzione al DB DR (utilizzando la modalità ASYNC).	1. Nel database di produzione, esegui i seguenti comandi per aggiornare i parametri.
	`db2 UPDATE DB CFG FOR <SID> USING HADR_LOCAL_HOST HOST1`
	`db2 UPDATE DB CFG FOR <SID> USING HADR_LOCAL_SVC <SID>_HADR_1`
	`db2 UPDATE DB CFG FOR <SID> USING HADR_REMOTE_HOST HOST2`
	`db2 UPDATE DB CFG FOR <SID> USING HADR_REMOTE_SVC <SID>_HADR_2`
	`db2 UPDATE DB CFG FOR <SID> USING HADR_REMOTE_INST db2<sid>`
	`db2 UPDATE DB CFG FOR <SID> USING HADR_TIMEOUT 120`
	`db2 UPDATE DB CFG FOR <SID> USING HADR_SYNCMODE ASYNC`
	`db2 UPDATE DB CFG FOR <SID> USING HADR_SPOOL_LIMIT 1000`
	`db2 UPDATE DB CFG FOR <SID> USING HADR_PEER_WINDOW 240`
	`db2 UPDATE DB CFG FOR <SID> USING indexrec RESTART logindexbuild ON`
	2. Nel database DR, esegui i seguenti comandi per aggiornare i parametri.
		Amministratore SAP Basis

Operazione	Description	Competenze richieste

	`db2 UPDATE DB CFG FOR <SID> USING HADR_LOCAL_HOST HOST2`
	`db2 UPDATE DB CFG FOR <SID> USING HADR_LOCAL_SVC <SID>_HADR_2`
	`db2 UPDATE DB CFG FOR <SID> USING HADR_REMOTE_HOST HOST1`
	`db2 UPDATE DB CFG FOR <SID> USING HADR_REMOTE_SVC <SID>_HADR_1`
	`db2 UPDATE DB CFG FOR <SID> USING HADR_REMOTE_INST db2<sid>`
	`db2 UPDATE DB CFG FOR <SID> USING HADR_TIMEOUT 120`
	`db2 UPDATE DB CFG FOR <SID> USING HADR_SYNCMODE ASYNC`
	`db2 UPDATE DB CFG FOR <SID> USING HADR_SPOOL_LIMIT 1000`
	`db2 UPDATE DB CFG FOR <SID> USING HADR_PEER_WINDOW 240`
	`db2 UPDATE DB CFG FOR <SID> USING indexrec RESTART logindexbuild ON`
	
	Questi parametri sono necessari per fornire informazioni relative all'HADR a entrambi i database. Nel database Db2, HADR viene attivato in base ai valori di ciascuno	

Operazione	Description	Competenze richieste

	dei parametri precedent emente impostati. Per ulteriori informazioni su questi parametri, consulta la documentazione IBM.
	3. Avviare innanzitutto HADR sul database di standby appena creato utilizzando il comando seguente.
	`db2 deactivate db <SID>`
	`db2 start hadr on db <SID> as standby`
	4. Avviare HADR sul database di produzione utilizzando il comando seguente.
	`db2 deactivate db <SID>`
	`db2 start hadr on db <SID> as primary`
	5. Verificate se i database Db2 di produzione e di standby sono sincronizzati e la spedizione dei log è in corso.
	Per monitorare lo stato della replica HADR, utilizzare il comando seguente. db2pd
	`db2pd -d <SID> -hadr`	

Operazione	Description	Competenze richieste
	Per ulteriori informazioni sul monitoraggio di HADR, consulta la documentazione IBM.	

Test delle attività di failover del DR

Operazione	Description	Competenze richieste
Pianifica i tempi di inattività dell'attività di produzione per il test DR.	Assicurati di pianificare i tempi di inattività aziendali richiesti nell'ambiente di produzion e per testare lo scenario di failover del DR.	Amministratore SAP Basis
Crea un utente di prova.	Crea un utente di test (o eventuali modifiche al test) che possa essere convalidato nell'host DR per confermare la replica dei log dopo il failover DR.	Amministratore SAP Basis
Sulla console, arresta le EC2 istanze di produzione.	In questa fase viene avviato lo spegnimento indesiderato per simulare uno scenario di emergenza.	Amministratore di sistema AWS
Scala l' EC2 istanza DR per soddisfare i requisiti.	Sulla EC2 console, modifica il tipo di istanza nella regione DR. 1. Arresta l'istanza: se l'istanza è in esecuzione, è necessario interromperla prima di poterne modificar	SAP Basis Admin

Operazione	Description	Competenze richieste

	e il tipo. Sulla EC2 console, seleziona l'istanza e scegli Stop.
	2. Modifica il tipo di istanza: sulla EC2 console, seleziona l'istanza e scegli Azioni, Impostazioni istanza, Modifica tipo di istanza. Seleziona il tipo di istanza che corrisponde all'istanza principale e scegli Applica.
	3. Avvia l'istanza: una volta completata la modifica del tipo di istanza, avvia l'istanza dalla EC2 console selezionando l'istanza e scegliendo Avvia.
	4. Per avviare il database Db2, utilizzate il seguente comando.
	`db2start`
	`db2 start HADR on db <SID> as standby`	

Operazione	Description	Competenze richieste

Avviare l'acquisizione.	Dal sistema DR (`host2`), avvia il processo di acquisizione e richiama il database DR come principale.
	`db2 takeover hadr on database <SID> by force`
	Facoltativamente, è possibile impostare i seguenti parametri per regolare automaticamente l'allocazione della memoria del database in base al tipo di istanza. Il `INSTANCE_MEMORY` valore può essere deciso in base alla porzione di memoria dedicata da allocare al database Db2.
	`db2 update db cfg for <SID> using INSTANCE_MEMORY <FIXED VALUE> IMMEDIATE;`
	`db2 get db cfg for <SID> | grep -i DATABASE_MEMORY AUTOMATIC IMMEDIATE;`
	`db2 update db cfg for <SID> using self_tuning_mem ON IMMEDIATE;`
	Verifica la modifica utilizzando i seguenti comandi.
	`db2 get db cfg for <SID> | grep -i MEMORY`
		Amministratore SAP Basis

Operazione	Description	Competenze richieste
	`db2 get db cfg for <SID> | grep -i self_tuning_mem`	
Avvia il server delle applicazioni per SAP nella regione DR.	Utilizzando l'AMI che hai creato per il sistema di produzione, avvia un nuovo server di applicazioni aggiuntivo nella regione DR.	Amministratore SAP Basis
Esegui la convalida prima di avviare l'applicazione SAP.	1. Convalida le voci e. `/etc/hosts` `/etc/fstab` 2. Montare `/sapmnt/<SID>/` sul sistema DR. 3. Verifica che il file system DR `/sapmnt/<SID>/` sia sincronizzato con la produzione. `/sapmnt/<SID>/` 4. Accedi all'`<sid>admutente`R3trans `-d`, esegui e verifica l'output nel `trans.log` file. Il `trans.log` file viene generato nella stessa posizione in cui è stato eseguito il `R3trans -d` comando.	Amministratore AWS, amministratore SAP Basis

Operazione	Description	Competenze richieste
Avvia l'applicazione SAP sul sistema DR.	Avviare l'applicazione SAP sul sistema DR utilizzando `<sid>adm` user. Utilizzat e il codice seguente, che XX rappresenta il numero di istanza del server SAP ABAP SAP Central Services (ASCS) e YY rappresenta il numero di istanza del server delle applicazioni SAP. `sapconrol -nr XX - function StartService <SID> sapconrol -nr XX - function StartSystem sapconrol -nr YY - function StartService <SID> sapconrol -nr YY - function StartSystem`	Amministratore SAP Basis
Eseguire la convalida SAP.	Viene eseguito come test DR per fornire prove o per verificar e il successo della replica dei dati nella regione DR.	Tecnico di test

Esegui attività di failback del DR

Operazione	Description	Competenze richieste
Avvia i server SAP e di database di produzione.	Sulla console, avvia le EC2 istanze che ospitano SAP e il database nel sistema di produzione.	Amministratore SAP Basis

Operazione	Description	Competenze richieste

Avvia il database di produzione e configura HADR.	Accedere al sistema di produzione (`host1`) e verificare che il DB sia in modalità di ripristino utilizzando il comando seguente.
	`db2start`
	`db2 start HADR on db P3V as standby`
	`db2 connect to <SID>`
	Verificate che lo stato HADR sia `connected` . Lo stato di replica dovrebbe essere. `peer`
	`db2pd -d <SID> -hadr`
	Se il database non è incoerente e non è in `connected` uno `peer` stato, potrebbero essere necessari un backup e un ripristino per sincronizzare il database (accesso`host1`) con il database attualmente attivo (`host2`nella regione DR). In tal caso, ripristina il backup del DB dal database nella regione `host2` DR al database nella regione `host1` di produzione.
		Amministratore SAP Basis

Operazione	Description	Competenze richieste
Esegui il failback del database nella regione di produzione.	In uno business-as-usual scenario normale, questo passaggio viene eseguito in un periodo di inattività programmato. Le applicazioni in esecuzione sul sistema DR vengono interrotte e il database viene riportato alla regione di produzione (Regione 1) per riprendere le operazioni dalla regione di produzione. 1. Accedere al server delle applicazioni SAP nella regione DR e interrompere l'applicazione SAP. 2. Smonta `/sapmnt/<SID>` dal sistema DR, assicurandoti che le modifiche vengano replicate in senso inverso sul sistema di produzione. `/sapmnt/<SID>` 3. Accedere al server del database (`host1`) nella regione di produzione ed eseguire l'acquisizione. `db2 takeover hadr on database <SID>` 4. Controlla lo stato HADR: `HADR_ROLE` dovrebbe essere acceso `host1` e	Amministratore SAP Basis

Operazione	Description	Competenze richieste
	PRIMARY StandBy acceso. host2 `db2pd -d <SID> -hadr`	
Esegui la convalida prima di avviare l'applicazione SAP.	1. Convalida le voci e. /etc/ hosts /etc/fstab 2. Montare /sapmnt/< SID>/ sul sistema di produzione. 3. Assicurati che sia sincroniz zato con il sistema DR/ sapmnt/<SID>/ . 4. Accedi all'<sid>admutenteR3trans -d, esegui e verifica l'output nel trans.log file. Il trans.log file viene generato nella stessa posizione in cui è stato eseguito il R3trans -d comando.	Amministratore AWS, amministratore SAP Basis

Operazione	Description	Competenze richieste

Avvia l'applicazione SAP.	1. Avviare l'applicazione SAP sul sistema di produzione utilizzando `<sid>adm` l'utente. Utilizzate il codice seguente, che `XX` rappresenta il numero di istanza del server SAP ASCS e `YY` rappresenta il numero di istanza del server delle applicazioni SAP.
	`sapconrol -nr XX -function StartService <SID>`
	`sapconrol -nr XX -function StartSystem`
	`sapconrol -nr YY -function StartService <SID>`
	`sapconrol -nr YY -function StartSystem`
	2. Per confermare che i server delle applicazioni sono disponibili, accedi a SAP ed esegui i controlli utilizzando SICK e le transazioni. SM51
		Amministratore SAP Basis

risoluzione dei problemi

Problema	Soluzione
File di registro e comandi chiave per la risoluzione dei problemi relativi all'HADR	• `db2 get db cfg | grep -i hadr`

Problema	Soluzione
	• `db2pd -d sid -hadr` • `Db2diag.log` (Questo file si trova generalmente all'interno della `db2dump` directory e il `db2dump` percorso è definito dal parametro.) `DIAGPATH`
Nota SAP per la risoluzione dei problemi HADR su Db2 UDB	Fare riferimento alla Nota SAP 1154013 - DB6: problemi del DB in ambiente HADR. (Sono necessarie le credenziali del portale SAP per accedere a questa nota.)

Risorse correlate

- Approcci di disaster recovery per database Db2 su AWS (post sul blog)
- SAP su AWS — IBM Db2 HADR con Pacemaker
- Procedura dettagliata per configurare la replica HADR tra database DB2
- Wiki Db2 HADR

Informazioni aggiuntive

Utilizzando questo modello, è possibile configurare un sistema di disaster recovery per un sistema SAP in esecuzione sul database Db2. In una situazione di emergenza, l'azienda dovrebbe essere in grado di continuare a rispettare i requisiti RTO (Recovery Time Objective) e RPO (Recovery Point Objective) definiti:

- L'RTO è il ritardo massimo accettabile tra l'interruzione del servizio e il ripristino del servizio. Questo determina ciò che viene considerato un intervallo di tempo accettabile in caso di indisponibilità del servizio.
- L'RPO è il periodo di tempo massimo accettabile dall'ultimo punto di ripristino dei dati. Questo determina ciò che si considera una perdita di dati accettabile tra l’ultimo punto di ripristino e l’interruzione del servizio.

Per informazioni FAQs relative all'HADR, consulta la nota SAP #1612105 - DB6: Domande frequenti su Db2 High Availability Disaster Recovery (HADR). (Sono necessarie le credenziali del portale SAP per accedere a questa nota.)

Configura una CI/CD pipeline per la migrazione del database utilizzando Terraform

Dott. Rahul Sharad Gaikwad, Ashish Bhatt, Aniket Dekate, Ruchika Modi, Tamilselvan P, Nadeem Rahaman, Aarti Rajput e Naveen Suthar, Amazon Web Services

Riepilogo

Questo modello prevede la creazione di una pipeline di integrazione e distribuzione continua (CI/CD) per la gestione delle migrazioni dei database in modo affidabile e automatizzato. Copre il processo di fornitura dell'infrastruttura necessaria, la migrazione dei dati e la personalizzazione delle modifiche allo schema utilizzando Terraform, che è uno strumento di infrastruttura come codice (IaC).

In particolare, il modello imposta una CI/CD pipeline per migrare un database Microsoft SQL Server locale su Amazon Relational Database Service (Amazon RDS) su. AWS Puoi anche utilizzare questo modello per migrare un database SQL Server che si trova su una macchina virtuale (VM) o in un altro ambiente cloud su Amazon RDS.

Questo modello affronta le seguenti sfide associate alla gestione e alla distribuzione dei database:

- Le implementazioni manuali dei database richiedono molto tempo, sono soggette a errori e mancano di coerenza tra gli ambienti.
- Il coordinamento del provisioning dell'infrastruttura, delle migrazioni dei dati e delle modifiche allo schema può essere complesso e difficile da gestire.
- Garantire l'integrità dei dati e ridurre al minimo i tempi di inattività durante gli aggiornamenti del database è fondamentale per i sistemi di produzione.

Questo modello offre i seguenti vantaggi:

- Semplifica il processo di aggiornamento e implementazione delle modifiche al database implementando una CI/CD pipeline per le migrazioni dei database. Ciò riduce il rischio di errori, garantisce la coerenza tra gli ambienti e riduce al minimo i tempi di inattività.
- Aiuta a migliorare l'affidabilità, l'efficienza e la collaborazione. Consente un time-to-market più rapido e tempi di inattività ridotti durante gli aggiornamenti del database.
- Ti aiuta ad adottare DevOps pratiche moderne per la gestione dei database, il che porta a una maggiore agilità, affidabilità ed efficienza nei processi di distribuzione del software.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Terraform 0.12 o versione successiva installato sul computer locale (per le istruzioni, consulta la documentazione di Terraform)
- Terraform AWS Provider versione 3.0.0 o successiva da HashiCorp (consulta il repository di questo provider) GitHub
- Politica sui privilegi minimi AWS Identity and Access Management (IAM) (vedi il post sul blog Tecniche per scrivere politiche IAM con privilegi minimi)

Architecture

Questo modello implementa la seguente architettura, che fornisce l'infrastruttura completa per il processo di migrazione del database.

In questa architettura:

- Il database di origine è un database SQL Server locale, su una macchina virtuale (VM) o ospitato da un altro provider di servizi cloud. Il diagramma presuppone che il database di origine si trovi in un data center locale.
- I data center locali AWS sono collegati tramite una VPN o una connessione. AWS Direct Connect Ciò fornisce comunicazioni sicure tra il database di origine e l' AWS infrastruttura.
- Il database di destinazione è un database Amazon RDS ospitato all'interno del cloud privato virtuale (VPC) con l'aiuto di una AWS pipeline di provisioning del database.
- AWS Database Migration Service (AWS DMS) replica il database locale su. AWS Viene utilizzato per configurare la replica del database di origine nel database di destinazione.

Il diagramma seguente mostra l'infrastruttura configurata con diversi livelli del processo di migrazione del database, che prevede il provisioning, la AWS DMS configurazione e la convalida.

In questo processo:

- La pipeline di convalida convalida tutti i controlli. La pipeline integrata passa alla fase successiva quando tutte le convalide necessarie sono state completate.
- La pipeline di provisioning del DB è composta da varie AWS CodeBuild fasi che eseguono azioni Terraform sul codice Terraform fornito per il database. Una volta completati questi passaggi, distribuisce le risorse nell'obiettivo. Account AWS
- La AWS DMS pipeline è composta da varie CodeBuild fasi che eseguono i test e quindi forniscono l' AWS DMS infrastruttura per eseguire la migrazione utilizzando IaC.

Tools (Strumenti)

Servizi AWS e strumenti

- AWS CodeBuildè un servizio di integrazione continua completamente gestito che compila codice sorgente, esegue test e produce pacchetti ready-to-deploy software.
- AWS CodePipelineè un servizio di distribuzione continua completamente gestito che consente di automatizzare le pipeline di rilascio per aggiornamenti rapidi e affidabili di applicazioni e infrastrutture.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale in. Cloud AWS
- Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è un servizio di storage di oggetti che offre scalabilità, disponibilità dei dati, sicurezza e prestazioni.
- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati all'interno Cloud AWS o tra combinazioni di configurazioni cloud e locali.

Altri servizi

- Terraform è uno strumento IaC HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.

Archivio di codice

Il codice per questo modello è disponibile nel GitHub Database Migration DevOps Framework utilizzando il repository di esempi Terraform.

Best practice

- Implementa test automatici per la migrazione del database per verificare la correttezza delle modifiche allo schema e l'integrità dei dati. Ciò include test unitari, test di integrazione e end-to-end test.
- Implementa una solida strategia di backup e ripristino per i tuoi database, soprattutto prima della migrazione. Ciò garantisce l'integrità dei dati e fornisce un'opzione di riserva in caso di guasti.
- Implementa una solida strategia di rollback per annullare le modifiche al database in caso di errori o problemi durante la migrazione. Ciò potrebbe comportare il ripristino a uno stato precedente del database o il ripristino di singoli script di migrazione.
- Imposta meccanismi di monitoraggio e registrazione per tenere traccia dell'avanzamento e dello stato delle migrazioni del database. Questo ti aiuta a identificare e risolvere rapidamente i problemi.

Epiche

Configura la tua workstation locale

Operazione	Description	Competenze richieste
Configura e configura Git sulla tua workstation locale.	Installa e configura Git sulla tua workstation locale seguendo le istruzioni nella documentazione di Git.	DevOps ingegnere
Crea una cartella di progetto e aggiungi i file dal GitHub repository.	1. Apri il GitHub repository per questo pattern. 2. Scegli Code per visualizz are le opzioni di clonazion e e copia l'URL fornito nella scheda HTTPS. 3. Crea una cartella per il tuo progetto sulla tua workstati on. 4. Apri un terminale e accedi a questa cartella. 5. Clona il GitHub repository:	DevOps ingegnere

Operazione	Description	Competenze richieste

	`git clone <github-repository-url>`
	
	`<github-repository-url>` dov'è l'URL che hai copiato nel passaggio 2.
	6. Quando la clonazione è completa, vai al repository clonato nella cartella del progetto:
	
	`cd <folder-name>/aws-terraform-db-migration-framework-samples`
	7. Apri questo progetto in un ambiente di sviluppo integrato (IDE) di tua scelta.	

Fornisci l'architettura di destinazione

Operazione	Description	Competenze richieste
Aggiorna i parametri richiesti.	Il `ssm-parameters.sh` file memorizza tutti i AWS Systems Manager parametri richiesti. Puoi configurare questi parametri con i valori personalizzati per il tuo progetto. Nella `setup/db-ssm-params` cartella sulla workstation locale, aprite il	DevOps ingegnere

Operazione	Description	Competenze richieste
	`ssm-parameters.sh` file e impostate questi parametri prima di eseguire la CI/CD pipeline.	
Inizializza la configurazione Terraform.	Nella `db-cicd-integration` cartella, inserisci il seguente comando per inizializzare la tua directory di lavoro che contiene i file di configurazione Terraform: `terraform init`	DevOps ingegnere
Visualizza l'anteprima del piano Terraform.	Per creare un piano Terraform , inserisci il seguente comando: `terraform plan -var-file ="terraform.sample"` Terraform valuta i file di configurazione per determina re lo stato di destinazione per le risorse dichiarate. Quindi confronta lo stato di destinazi one con lo stato attuale e crea un piano.	DevOps ingegnere
Verifica il piano.	Rivedi il piano e conferma che configuri l'architettura richiesta nel tuo obiettivo Account AWS.	DevOps ingegnere

Operazione	Description	Competenze richieste
Distribuire la soluzione.	1. Inserisci il seguente comando per applicare il piano: `terraform apply -var-file="terraform.sample"` 2. Immettere `yes` per confermare. Terraform crea, aggiorna o distrugge l'infrastruttura per raggiungere lo stato di destinazione dichiarato nei file di configurazione. Per ulteriori informazioni sulla sequenza, vedere la sezione Architettura di questo modello.	DevOps ingegnere

Verifica della distribuzione

Operazione	Description	Competenze richieste
Convalida la distribuzione.	Verifica lo stato della `db-cicd-integration` pipeline per confermare che la migrazione del database sia completa. 1. Accedi a Console di gestione AWS, quindi apri la AWS CodePipeline console.	DevOps ingegnere

Operazione	Description	Competenze richieste
	2. Nel riquadro di navigazione, scegli Pipelines. 3. Scegliete la `db-cicd-integration` pipeline. 4. Verifica che l'esecuzione della pipeline sia stata completata correttamente.	

Pulisci l'infrastruttura dopo l'uso

Operazione	Description	Competenze richieste
Pulisci l'infrastruttura.	1. Una volta completato il progetto, pulisci l'infrastruttura che hai creato utilizzando il comando: `terraform destroy --var-file=terraform.sample` 2. Immettere `yes` per confermare.	DevOps ingegnere

Risorse correlate

AWS documentazione

- Iniziare con un prodotto Terraform

Documentazione Terraform

- Installazione di Terraform

- Configurazione del backend Terraform
- Documentazione Terraform Provider AWS

Configura un cluster di failover di Microsoft SQL Server su Amazon EC2 utilizzando FSx per Windows File Server

Sweta Krishna e Ramesh Babu Donti, Amazon Web Services

Riepilogo

L'edizione Microsoft SQL Server Standard con un'istanza cluster di failover (FCI) può fornire un'alternativa più economica a SQL Server Enterprise. La configurazione di SQL FCI richiede lo storage di file condiviso tra i nodi e Amazon FSx for Windows File Server fornisce uno storage completamente gestito che si replica automaticamente in modo sincrono tra le zone di disponibilità. Amazon FSx riduce i costi di storage utilizzando la deduplicazione dei dati integrata per le condivisioni di file generiche, che elimina la necessità di mantenere soluzioni di terze parti. Amazon supporta FSx anche quanto segue:

- Paghi solo per ciò che usi senza costi o impegni iniziali.
- Configura FCI manualmente FSx come storage condiviso.
- Utilizzalo FSx come testimone della condivisione dei file per il tuo cluster SQL.
- Amazon FSx per Windows File Server supporta Server Message Block (SMB) 3.0 per le condivisioni di file sempre disponibili, rendendolo adatto per le distribuzioni FCI di SQL Server.

Prerequisiti e limitazioni

Prerequisiti

- Account AWSAttivo.
- Autorizzazioni per creare e gestire risorse Amazon Virtual Private Cloud (Amazon VPC), istanze Amazon Elastic Compute Cloud (EC2Amazon), gruppi di sicurezza AWS Identity and Access Management e ruoli (IAM).
- AWS Managed Microsoft AD o il tuo Active Directory locale.
- Un utente del dominio Active Directory con l'autorizzazione necessaria per configurare un cluster di failover.
- Regole dei gruppi di sicurezza per le porte SQL Server FCI e Microsoft Active Directory per una connettività ibrida sicura.
- Un account di servizio in Active Directory per SQL Server configurato con le autorizzazioni appropriate tra i nodi SQL.

- Amazon FSx per Windows File Server in un cluster di failover.
- File binari di installazione di SQL Server.

Limitazioni

- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- Amazon EC2 per Windows Server 2012 R2 o versione successiva
- Amazon FSx per Windows File Server con tutte le versioni correnti di Windows Server
- Amazon FSx for NetApp ONTAP come alternativa allo storage condiviso
- SQL Server 2012/2016/2019/2022

Architecture

Stack tecnologico

- Amazon EC2
- File server Amazon FSx per Windows
- Amazon VPC
- AWS Directory Service
- AWS Systems Manager
- IAM

Architettura Target

Il diagramma seguente mostra l'architettura di alto livello di Microsoft SQL Server FCI su Amazon utilizzando EC2 Amazon FSx for Windows File Server.

Infrastruttura di rete

- Amazon VPC fornisce un contenitore di rete che si estende su tre zone di disponibilità.

- Le sottoreti private forniscono sottoreti isolate in ogni zona di disponibilità per la distribuzione delle risorse.

Livello di elaborazione

- Amazon EC2 contiene un nodo 1 del cluster SQL Server, distribuito nella Zona di disponibilità 1 come parte del Windows Server Failover Cluster (WSFC).
- Amazon EC2 contiene un nodo 2 del cluster SQL Server, distribuito nella Zona di disponibilità 2 come parte del WSFC.
- Il cluster WSFC collega entrambi i nodi SQL Server per la funzionalità di failover.

Livello di archiviazione per Amazon FSx per Windows File Server

FSx Implementazione Multi-AZ (che copre le zone di disponibilità 1 e 2)

- Un FSx file system primario nella Zona di disponibilità 1 ospita dati e file di registro attivi di SQL Server.
- Un FSx file system secondario nella Zona di disponibilità 2 offre funzionalità di failover automatico.
- Una condivisione di file SMB condivisa (\\ fsx.domain\ sqlshare), accessibile da entrambi i nodi del cluster per i database SQL Server.

Implementazione Single-AZ (in) FSx AZ3

- Amazon FSx File Server Witness in Availability Zone 3 funge da testimone del quorum del cluster.
- Il file share witness (`\\fsx.domain\witness`) mantiene il quorum del cluster e previene gli scenari split-brain.

Servizi di directory

- AWS Managed Microsoft AD fornisce i servizi di autenticazione e dominio di Windows necessari per la funzionalità del cluster.

Funzionalità di elevata disponibilità

- I componenti Multi-AZ offrono tolleranza agli errori in tutte le zone di disponibilità.
- FSx il file server di standby fornisce il failover automatico in caso di guasto del server primario.

- File share witness consente la gestione del quorum del cluster nella zona di disponibilità 3 per garantire il corretto funzionamento del cluster in caso di guasti.
- Domain è integrato con AWS Managed Microsoft AD per un'autenticazione Windows senza interruzioni.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon FSx fornisce file system che supportano i protocolli di connettività standard del settore e offrono disponibilità e replica elevate in tutto il mondo. Regioni AWS
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.
- AWS Directory Service for Microsoft Active Directoryconsente ai carichi di lavoro e alle risorse compatibili con le directory di AWS utilizzare Microsoft Active Directory in. Cloud AWS
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Systems Managerti aiuta a gestire le applicazioni e l'infrastruttura in esecuzione in. Cloud AWS Semplifica la gestione delle applicazioni e delle risorse, riduce i tempi di rilevamento e risoluzione dei problemi operativi e aiuta a gestire le AWS risorse in modo sicuro su larga scala.

Best practice

- Posiziona le istanze di database in sottoreti private per proteggerle dall'accesso pubblico da Internet, pur consentendo loro di connettersi ed eseguire aggiornamenti. Servizi AWS
- Per le best practice generali, consulta Best practice per la distribuzione di Microsoft SQL Server su Amazon EC2.
- Da utilizzare PowerShell per amministrare il tuo file server Amazon FSx for Windows, consulta Amministrazione FSx per i file system Windows.

Epiche

Crea e configura EC2 nodi Amazon per SQL Server

Operazione	Description	Competenze richieste
Aggiungi nomi e tag.	1. Nel Console di gestione AWS, vai su Amazon EC2. 2. Scegliere Launch Instance (Avvia istanza). 3. Aggiungi i tag richiesti.	DBA
Scegli un AMI Windows.	Scegli un'Amazon Machine Image (AMI) per Windows che soddisfi i requisiti di SQL Server.	DBA
Selezione di un tipo di istanza.	Seleziona un tipo di EC2 istanza Amazon che soddisfi i tuoi requisiti.	DBA
Usa una key pair.	Puoi usare una key pair per connetterti in modo sicuro alla tua istanza. Assicurati di avere accesso alla coppia di key pair selezionata prima di avviare l'istanza.	DBA
Configurazione delle impostazioni di rete.	1. Scegli il VPC che hai usato per configurare Active Directory. 2. Scegli il VPC e la sottorete privata. 3. Scegli un gruppo di sicurezza esistente. Se non ne hai uno, scegli Crea gruppo di sicurezza.	DBA

Operazione	Description	Competenze richieste
Configura impostazioni di rete avanzate.	1. Scegli Impostazioni di rete avanzate. 2. Seleziona l'IP secondario. 3. Seleziona la casella di controllo Assegna automaticamente per assegnare l'IP secondario dalla sottorete.	DBA
Configura l'archiviazione.	Configura lo spazio di archiviazione totale richiesto e scegli il tipo di archiviazione richiesto.	DBA
Configura i dettagli avanzati e avvia l'istanza.	1. Scegli il dominio Active Directory dall'elenco delle directory. 2. Scegli un ruolo IAM a cui siano associate `AmazonSSMDirectoryServiceAccess` le policy `AmazonSSMManagedInstanceCore` e. 3. (Facoltativo) Configura altre opzioni disponibili. 4. Avviare l'istanza.	DBA
Crea il nodo 2.	Ripeti questi passaggi per creare e configurare il nodo 2.	DBA

Installa e configura il cluster di failover di Windows Server sui nodi 1 e 2

Operazione	Description	Competenze richieste
Accedere al nodo 1.	Accedi all' EC2 istanza Amazon di Windows come amministratore.	DBA
Installa le funzionalità FCI sul nodo 1.	1. In PowerShell, esegui lo script seguente: `Install-WindowsFeature -Name Failover-Clustering -IncludeManagementTools` 1. Dopo l'installazione, verifica e installa eventuali aggiornamenti di Windows in sospeso. 2. Riavvia l’istanza.	DBA
Accedere al nodo 2.	Accedi all' EC2 istanza Amazon di Windows come amministratore.	DBA
Installa le funzionalità FCI sul nodo 2.	1. In PowerShell, esegui lo script seguente: `Install-WindowsFeature -Name Failover-Clustering -IncludeManagementTools` 1. Dopo l'installazione, verifica e installa eventuali	DBA

Operazione	Description	Competenze richieste
	aggiornamenti di Windows in sospeso. 2. Riavvia l’istanza.	

Operazione	Description	Competenze richieste
Aggiungi nodi al cluster.	1. Usa Remote Desktop per connetterti al nodo 1 e scegli Start server manager. 2. Nel menu Strumenti, scegli Failover Cluster Manager. 3. Nel riquadro Gestione del cluster di failover in Gestione, scegli Crea cluster per aprire la procedura guidata per la creazione del cluster. 4. Nella pagina Prima di iniziare scegliere Avanti. 5. Se viene visualizzata la pagina Seleziona server, nel campo Inserisci nome, scegli Sfoglia e cerca i tuoi server, quindi scegli Aggiungi. Ripeti questo passaggio per ogni server che desideri aggiungere, quindi scegli Avanti. 6. Nella pagina Punto di accesso per l'amminis trazione del cluster, scegli il cluster, quindi scegli Avanti. 7. La pagina di conferma mostra il nome del cluster, il nodo e altre informazi oni sul dominio. Controlla questi dettagli, quindi scegli Avanti.	DBA

Operazione	Description	Competenze richieste
	8. Nella pagina di riepilogo , verifica che la FCI sia stata creata correttamente. Se sono presenti avvisi o errori, visualizza l'output di riepilogo o scegli Visualizza rapporto. 9. Scegli Fine. 10. Quando il cluster è configurato, il sistema visualizza Hai completato la procedura guidata per la creazione del cluster. 11. Ripetere questi passaggi per il nodo 2.	

Operazione	Description	Competenze richieste
Porta il cluster online.	Per portare il cluster online, aggiorna gli indirizzi IP statici di entrambi i nodi: 1. Seleziona il nome dell'oggetto del cluster. 2. Apri il menu contestuale (fai clic con il pulsante destro del mouse) e scegli Proprietà. 3. Seleziona la sottorete e un indirizzo IP secondario non utilizzato dall'istanza. EC2 4. Dopo aver aggiornato l'indirizzo IP secondario, apri il menu contestuale (fai clic con il pulsante destro del mouse) e seleziona il nome dell'oggetto del cluster. 5. Scegli Porta online.	DBA
Convalida il cluster.	Accedi a Failover Cluster Manager e verifica che le risorse principali del cluster siano online.	DBA

Installa SQL Server sui nodi 1 e 2

Operazione	Description	Competenze richieste
Accedere al server.	Accedi all' EC2 istanza Amazon come amministratore.	DBA

Operazione	Description	Competenze richieste
Monta i binari SQL.	1. Avvia il centro di installaz ione di SQL Server, quindi scegli Installazione. 2. Scegli Nuova installazione del cluster di failover di SQL Server. 3. Segui le istruzioni fino ai file di installazione. 4. Nella finestra di dialogo Regole globali, verificate che tutti i controlli abbiano avuto esito positivo. 5. Nella finestra di dialogo Installa le regole del cluster di failover, verificate che tutti i controlli abbiano avuto esito positivo. Rispondi a eventuali avvisi prima di procedere. 6. Nella finestra di dialogo di selezione delle funzional ità, seleziona le caselle di controllo per la connettiv ità dei servizi del motore di database e degli strumenti client. 7. Nella finestra di dialogo Feature rules, verificate che tutte le regole siano state rispettate. 8. Nella finestra di dialogo di configurazione dell'istanza,	

Operazione	Description	Competenze richieste
	inserisci il nome della rete SQL Server. 9. Nel gruppo di risorse Cluster, specificare il nome del gruppo di risorse. Seleziona un gruppo dall'elenco a discesa o inserisci un nome personalizzato. 10. Nella finestra di dialogo di configurazione della rete del cluster, seleziona IPv4e deseleziona DHCP. Fornisci un indirizzo IP privato secondario per ogni nodo. 11. Nella finestra di dialogo di configurazione del server: • Nella scheda Account di servizio, fornisci i dettagli richiesti per il nome dell'account e le credenziali di SQL Service per SQL Server. • Verificare che il tipo di avvio per l'agente e il motore di database sia impostato su Manuale. • Seleziona la casella di controllo Concedi il privilegio Esegui attività di manutenzione del volume al servizio del motore di database SQL Server.	

Operazione	Description	Competenze richieste
	Ciò consente l'inizial izzazione istantanea dei file per SQL Server. 12. Nella finestra di dialogo di configurazione del motore di database: • Nella scheda Configura zione del server in modalità di autentica zione Windows, verifica che sia selezionata la modalità di autentica zione. • Nella scheda Directory di dati, specifica la posizione del FSx percorso Amazon per dati, log, tempdb e file di backup (ad esempio, per file e `\\fsx.domain \sqlshare\data \ \fsx.domain\sqlsh are\logs` per `.mdf` file). `.ldf` 13. Nella finestra di dialogo delle regole di configura zione delle funzional ità, verifica che tutti i controlli abbiano avuto esito positivo, quindi scegli Avanti. 14. Nella finestra di dialogo Pronto per l'installazione,	

Operazione	Description	Competenze richieste
	verificate tutte le impostazioni di configurazione. 15. Scegliete Installa per procedere con l'installazione. 16. Nella finestra di dialogo Completa, scegliete Chiudi.	

Operazione	Description	Competenze richieste
Aggiungere il nodo 2 al cluster di failover.	1. Accedere al centro di installazione di SQL Server. 2. Scegli Installazione. 3. Scegli Aggiungi nodo a un cluster di failover di SQL Server e segui le istruzioni fino alla configurazione del nodo Cluster. 4. Nella finestra di dialogo di configurazione del nodo Cluster, seleziona IPv4e deseleziona DHCP. 5. Aggiungi un indirizzo IP per il nodo 2. 6. In Account di servizio, verifica che i dettagli corrispondano per i nodi 1 e 2. 7. Fornisci le credenziali per gli account di servizio SQL Server corrispondenti. 8. Nella finestra di dialogo Regole delle funzional ità, verificate che tutti i controlli abbiano avuto esito positivo. 9. Scegliete Avanti e seguite le istruzioni rimanenti. 10. Scegli Installa per procedere con l'install azione.	DBA

Operazione	Description	Competenze richieste
	11. Nella finestra di dialogo Completa, scegliete Chiudi.	

Configurare Amazon FSx File Share Witness

Operazione	Description	Competenze richieste
Configura le impostazioni del quorum.	1. Usa RDP per connetter ti alla tua EC2 istanza Amazon. 2. Vai a Failover Cluster Manager. 3. Apri il menu contestuale (fai clic con il pulsante destro del mouse) sul cluster e scegli Altre azioni. 4. Scegli Configura le impostazioni del quorum del cluster. 5. Configura le impostazioni del quorum.	DBA
Recupera i dettagli DNS.	Nella FSx console Amazon, scegli Managed AD e poi Allega. Il DNS dovrebbe avere il seguente formato: `\\example.example.net\share`	DBA
Configura il file share witness.	Scegli il percorso di condivisi one dei FSx file di Amazon e poi Finish.	DBA

Risorse correlate

AWS resources

- Amazon FSx per Windows File Server (video)
- Approfondimento su Amazon FSx per Windows File Server (video)
- Come distribuire un cluster di failover di SQL Server con Amazon EBS Multi-Attach su Windows Server (post di blog)AWS
- Semplifica le distribuzioni ad alta disponibilità di Microsoft SQL Server utilizzando Amazon FSx per Windows File Server (AWS post sul blog)
- Implementazioni ad alta disponibilità di SQL Server con Amazon FSx for NetApp ONTAP (AWS post sul blog)
- Utilizzo FSx per Windows File Server con Microsoft SQL Server
- Che cos'è FSx per Windows File Server?

Altre risorse

- Creare un cluster di failover

Informazioni aggiuntive

Configurazione del file share witness

Assicurati di essere connesso al file system da entrambi i nodi aggiungendo regole nel gruppo di FSx sicurezza di Amazon che consentono le connessioni in entrata. La porta SMB dovrebbe essere consentita. Ad esempio, se il nome DNS è`\\example.example.com\share`, usa. `\\example.example.com\share` Utilizza lo stesso valore per il controllo della condivisione dei file nel cluster di disponibilità Always On. Completa i seguenti passaggi per configurare il file share witness:

1. Usa RDP per connetterti alla tua EC2 istanza Amazon.
2. Vai a Failover Cluster Manager.
3. Apri il menu contestuale (fai clic con il pulsante destro del mouse) e scegli Altre azioni.
4. Scegli Configura le impostazioni del quorum del cluster.
5. Scegli Next (Successivo).

6. Seleziona la configurazione Quorum e configura un file share witness.
7. Fornisci il nome DNS.
8. Esamina il riepilogo, quindi scegli Fine. Il file share witness dovrebbe essere online nella sezione Risorse principali del cluster.

Configurare un' HA/DR architettura per Oracle E-Business Suite su Amazon RDS Custom con un database di standby attivo

Simon Cunningham, Jaydeep Nandy e Nitin Saxena, Amazon Web Services

Riepilogo

Questo modello descrive come progettare una soluzione Oracle E-Business su Amazon Relational Database Service (Amazon RDS) Custom per l'alta disponibilità (HA) e il disaster recovery (DR) configurando un database di replica di lettura personalizzato Amazon RDS in un'altra zona di disponibilità di Amazon Web Services (AWS) e convertendolo in un database di standby attivo. La creazione della replica di lettura personalizzata di Amazon RDS è completamente automatizzata tramite la Console di gestione AWS.

Questo modello non descrive i passaggi per aggiungere livelli di applicazione aggiuntivi e file system condivisi, che possono anche far parte di un'architettura. HA/DR Per ulteriori informazioni su questi argomenti, vedere le seguenti note di supporto Oracle: 1375769.1, 1375670.1 e 1383621.1 (sezione 5, Opzioni di clonazione avanzate). (L'accesso richiede un account Oracle Support).

Per migrare il sistema E-Business Suite a un'architettura single-tier, Single-AZ su Amazon Web Services (AWS), consulta lo schema Migrate Oracle E-Business Suite to Amazon RDS Custom.

Oracle E-Business Suite è una soluzione Enterprise Resource Planning (ERP) per automatizzare processi a livello aziendale come quelli finanziari, delle risorse umane, delle catene di approvvigionamento e della produzione. Ha un'architettura a tre livelli: client, applicazione e database. In precedenza, dovevi eseguire il database E-Business Suite su un'istanza Amazon Elastic Compute Cloud (Amazon EC2) autogestita, ma ora puoi trarre vantaggio da Amazon RDS Custom.

Prerequisiti e limitazioni

Prerequisiti

- Un'installazione esistente di E-Business Suite su Amazon RDS Custom; vedi lo schema Migrare Oracle E-Business Suite ad Amazon RDS Custom
- Se desideri modificare la replica di lettura in modalità di sola lettura e utilizzarla per trasferire i report allo standby, una licenza per database Oracle Active Data Guard (consulta il listino prezzi commerciale di Oracle Technology)

Limitazioni

- Limitazioni e configurazioni non supportate per i database Oracle su Amazon RDS Custom
- Limitazioni associate alle repliche di lettura di Amazon RDS Custom for Oracle

Versioni del prodotto

Per le versioni e le classi di istanze di Oracle Database supportate da Amazon RDS Custom, consulta Requisiti e limitazioni per Amazon RDS Custom for Oracle.

Architecture

Il diagramma seguente illustra un'architettura rappresentativa per E-Business Suite su AWS che include più zone di disponibilità e livelli di applicazione in una configurazione. active/passive Il database utilizza un'istanza DB personalizzata di Amazon RDS e una replica di lettura Amazon RDS Custom. La replica di lettura utilizza Active Data Guard per la replica in un'altra zona di disponibilità. È inoltre possibile utilizzare la replica di lettura per scaricare il traffico di lettura sul database principale e per scopi di reporting.

Per ulteriori informazioni, consulta Lavorare con le repliche di lettura per Amazon RDS Custom for Oracle nella documentazione di Amazon RDS.

La replica di lettura personalizzata di Amazon RDS viene creata per impostazione predefinita come montata. Tuttavia, se desideri trasferire alcuni carichi di lavoro di sola lettura sul database di standby per ridurre il carico sul database principale, puoi modificare manualmente la modalità delle repliche montate in sola lettura seguendo i passaggi nella sezione Epics. Un tipico caso d'uso in questo caso sarebbe quello di eseguire i report dal database di standby. Il passaggio alla modalità di sola lettura richiede una licenza di database in standby attiva.

Quando crei una replica di lettura su AWS, il sistema utilizza il broker Oracle Data Guard sotto copertura. Questa configurazione viene generata automaticamente e configurata in modalità Maximum Performance come segue:

```
DGMGRL> show configuration
Configuration - rds_dg
  Protection Mode: MaxPerformance
  Members:
  vis_a - Primary database
    vis_b - Physical standby database
Fast-Start Failover: DISABLED
```

```
Configuration Status:
SUCCESS   (status updated 58 seconds ago)
```

Tools (Strumenti)

Servizi AWS

- Amazon RDS Custom for Oracle è un servizio di database gestito per applicazioni legacy, personalizzate e confezionate che richiedono l'accesso al sistema operativo e all'ambiente di database sottostanti. Automatizza le attività e le operazioni di amministrazione del database, consentendo al contempo, in qualità di amministratore di database, di accedere e personalizzare l'ambiente di database e il sistema operativo.

Altri strumenti

- Oracle Data Guard è uno strumento che consente di creare e gestire database Oracle standby. Questo modello utilizza Oracle Data Guard per configurare un database di standby attivo su Amazon RDS Custom.

Epiche

Creare una replica di lettura

Operazione	Description	Competenze richieste
Crea una replica di lettura dell'istanza database personalizzata di Amazon RDS.	Per creare una replica di lettura, segui le istruzioni nella documentazione di Amazon RDS e usa l'istanza database personalizzata di Amazon RDS che hai creato (consulta la sezione Prerequisiti) come database di origine. Per impostazione predefinita, la replica di lettura personalizzata di Amazon RDS viene creata come standby fisico e si	DBA

Operazione	Description	Competenze richieste
	trova nello stato montato. Ciò è intenzionale per garantire la conformità con la licenza Oracle Active Data Guard. Segui i passaggi successivi per convertire la replica di lettura in modalità di sola lettura.	

Cambia la replica di lettura in uno standby attivo di sola lettura

Operazione	Description	Competenze richieste

Connect alla replica di lettura personalizzata di Amazon RDS.	Usa i seguenti comandi per convertire il tuo database di standby fisico in un database di standby attivo.
	
	Important
	Questi comandi richiedono una licenza Oracle Active Standby. Per ottenere una licenza, contatta il tuo rappresentante Oracle.
	
	`$ sudo su - rdsdb`
	`-bash-4.2$ sql`
	`SQL> select process,s`
	`tatus,sequence# from v`
	`$managed_standby;`
		DBA

Operazione	Description	Competenze richieste
	PROCESS STATUS SEQUENCE# --------- ------------ ---------- ARCH CLOSING 3956 ARCH CONNECTED 0 ARCH CLOSING 3955 ARCH CLOSING 3957 RFS IDLE 0 RFS IDLE 3958 MRP0 APPLYING_LOG 3958 SQL> select name, database_role, open_mode from v $database; NAME DATABASE_ ROLE OPEN_MODE --------- --------- ------- --------- ----------- VIS PHYSICAL STANDBY MOUNTED SQL> alter database recover managed standby database cancel; Database altered. Open the standby database SQL> alter database open; Database altered. SQL> select name, database_role,	

Operazione	Description	Competenze richieste

	`open_mode from v $database;`
	
	`NAME DATABASE_ ROLE OPEN_MODE --------- --------- ------- --------- ----------- VIS PHYSICAL STANDBY READ ONLY`	

Avvia il ripristino dei file multimediali con log apply in tempo reale.	Per abilitare la funzionalità di applicazione dei log in tempo reale, utilizzate i seguenti comandi. Questi converton o e convalidano lo standby (read replica) come database in standby attivo, in modo da poter connettere ed eseguire query di sola lettura. `SQL> alter database recover managed standby database using current logfile disconnect from session; Database altered`	DBA

Operazione	Description	Competenze richieste

Controlla lo stato del database.	Per verificare lo stato del database, utilizzare il seguente comando.
	`SQL> select name, database_role, open_mode from v$database;`
	`NAME DATABASE_ROLE OPEN_MODE`
	`--------- ---------- ---------- VIS PHYSICAL STANDBY READ ONLY WITH APPLY`
		DBA

Operazione	Description	Competenze richieste

Seleziona la modalità Redo Apply.	Per controllare la modalità Redo Apply, utilizzate il seguente comando.
	
	`SQL> select process,s tatus,sequence# from v $managed_standby;`
	`PROCESS STATUS SEQUENCE#`
	`--------- ------------ ----------`
	`ARCH CLOSING 3956`
	`ARCH CONNECTED 0`
	`ARCH CLOSING 3955`
	`ARCH CLOSING 3957`
	`RFS IDLE 0`
	`RFS IDLE 3958`
	`MRP0 APPLYING_LOG 3958`
	
	`SQL> select open_mode from v$database;`
	`OPEN_MODE`
	`--------------------`
	`READ ONLY WITH APPLY`
		DBA

Risorse correlate

- Esegui la migrazione di Oracle E-Business Suite ad Amazon RDS Custom (AWS Prescriptive Guidance)
- Utilizzo di Amazon RDS Custom (documentazione Amazon RDS)

- Utilizzo delle repliche di lettura per Amazon RDS Custom for Oracle (documentazione Amazon RDS)
- Amazon RDS Custom per Oracle: nuove funzionalità di controllo nell'ambiente di database (blog AWS News)
- Migrazione di Oracle E-Business Suite su AWS (white paper AWS)
- Architettura di Oracle E-Business Suite su AWS (white paper AWS)

Trasmetti dati da IBM Db2, SAP, Sybase e altri database a MongoDB Atlas su AWS

Battulga Purevragchaa e Igor Alekseev, Amazon Web Services

Babu Srinivasan, MongoDB

Riepilogo

Questo modello descrive i passaggi per la migrazione dei dati da IBM Db2 e altri database come database mainframe e Sybase a MongoDB Atlas su. Cloud AWS Viene utilizzato AWS Glueper accelerare la migrazione dei dati verso MongoDB Atlas.

Il modello accompagna la guida Migrating to MongoDB Atlas sul AWS sito web Prescriptive Guidance. AWS Fornisce le fasi di implementazione per uno degli scenari di migrazione descritti in tale guida. Per ulteriori scenari di migrazione, consulta i seguenti modelli sul sito Web AWS Prescriptive Guidance:

- Migrare un ambiente MongoDB ospitato autonomamente su MongoDB Atlas su AWS
- Migrare database relazionali su MongoDB Atlas su AWS

Il modello è destinato ai partner e AWS agli utenti di AWS Managed Services.

Prerequisiti e limitazioni

Prerequisiti

- Un database di origine come SAP, Sybase, IBM Db2 e altri per la migrazione a MongoDB Atlas.
- Familiarità con database come SAP, Sybase, IBM Db2, MongoDB Atlas e. Servizi AWS

Versioni del prodotto

- MongoDB versione 5.0 o successiva.

Architecture

Il diagramma seguente illustra il caricamento e lo streaming di dati in batch utilizzando AWS Glue Studio Amazon Kinesis Data Streams e MongoDB Atlas.

Questa architettura di riferimento utilizza AWS Glue Studio per creare pipeline di estrazione, trasformazione e caricamento (ETL) per migrare i dati su MongoDB Atlas. An Crawler di AWS Glue si integra con MongoDB Atlas per facilitare la governance dei dati. I dati possono essere trasferiti in batch o trasmessi in streaming su MongoDB Atlas utilizzando Amazon Kinesis Data Streams.

Caricamento dati in batch

Per ulteriori informazioni sulla migrazione dei dati in batch, consulta il post del AWS blog Componi i tuoi lavori ETL per MongoDB Atlas con. AWS Glue

Streaming di dati

Per le architetture di riferimento di MongoDB Atlas che supportano diversi scenari di utilizzo, consulta Migrating to MongoDB Atlas sul sito Web Prescriptive Guidance. AWS AWS

Tools (Strumenti)

AWS Glue• è un servizio ETL completamente gestito. Ti aiuta a classificare, pulire, arricchire e spostare i dati in modo affidabile tra archivi di dati e flussi di dati.

• Amazon Kinesis Data Streams ti aiuta a raccogliere ed elaborare grandi flussi di record di dati in tempo reale.

• MongoDB Atlas è un database as a service (DBaaS) completamente gestito per l'implementazione e la gestione di database MongoDB nel cloud.

Best practice

Per le linee guida, consulta la Guida alle migliori pratiche per MongoDB nel repository MongoDB. GitHub

Poemi epici

Scoperta e valutazione

Operazione	Description	Competenze richieste
Determina la dimensione del cluster.	Stimate la dimensione del set di lavoro utilizzan	MongoDB DBA, architetto dell'applicazione

Operazione	Description	Competenze richieste
	do le informazioni fornite `db.stats()` per lo spazio totale dell'indice. Si supponga che si acceda frequente mente a una percentuale dello spazio dati. In alternati va, puoi stimare i requisiti di memoria in base alle tue ipotesi. Questa operazion e dovrebbe richiedere circa una settimana. Per ulteriori informazioni ed esempi su questa e le altre storie di questa epopea, consulta i collegamenti nella sezione Risorse correlate.	

Operazione	Description	Competenze richieste
Stima dei requisiti di larghezza di banda della rete.	Per stimare i requisiti di larghezza di banda di rete, moltiplica la dimensione media dei documenti per il numero di documenti inviati al secondo. Considerate come base il traffico massimo che qualsiasi nodo del cluster sarà in grado di sopportare. Per calcolare le velocità di trasferimento dei dati a valle dal cluster alle applicazioni client, utilizza la somma del totale dei documenti restituiti in un periodo di tempo. Se le tue applicazioni leggono da nodi secondari, dividi questo numero di documenti totali per il numero di nodi che possono eseguire operazioni di lettura. Per trovare la dimensione media dei documenti per un database, utilizzate il `db.stats().avgObjSize` comando. Questa operazione richiede in genere un giorno.	MongoDB DBA
Seleziona il livello Atlas.	Segui le istruzioni nella documentazione di MongoDB per selezionare il livello di cluster Atlas corretto.	MongoDB DBA
Piano di cutover.	Piano per il completamento dell'applicazione.	MongoDB DBA, architetto dell'applicazione

Configura un nuovo ambiente MongoDB Atlas su AWS

Operazione	Description	Competenze richieste
Crea un nuovo cluster MongoDB Atlas su. AWS	In MongoDB Atlas, scegli Crea un cluster e AWS seleziona come fornitore di servizi cloud.	MongoDB DBA
Configurazione selettiva Regioni AWS e globale del cluster.	Seleziona dall'elenco delle opzioni disponibili Regioni AWS per il tuo cluster Atlas. Se necessario, configura i cluster globali.	MongoDB DBA
Seleziona il livello del cluster.	Seleziona il livello di cluster preferito. La selezione del livello determina fattori quali memoria, storage e specifiche IOPS.	MongoDB DBA
Configura impostazioni aggiuntive del cluster.	Configura impostazioni cluster aggiuntive come la versione di MongoDB, il backup e le opzioni di crittografia. Per ulteriori informazioni su queste opzioni, consulta la sezione Risorse correlate.	MongoDB DBA

Configura sicurezza e conformità

Operazione	Description	Competenze richieste
Configura l'elenco di accesso.	Per connettersi al cluster Atlas, è necessario aggiunger e una voce all'elenco di accesso del progetto. Atlas utilizza Transport Layer	MongoDB DBA

Operazione	Description	Competenze richieste
	Security (TLS)/Secure Sockets Layer (SSL) per crittografare le connessioni al cloud privato virtuale (VPC) per il database. Per impostare l'elenco di accesso al progetto e per ulteriori informazioni sulle storie di questa epopea, consulta i collegamenti nella sezione Risorse correlate.	
Autentica e autorizza gli utenti.	È necessario creare e autenticare gli utenti del database che accederanno ai cluster MongoDB Atlas. Per accedere ai cluster di un progetto, gli utenti devono appartenere a quel progetto e possono appartenere a più progetti. Puoi anche abilitare l'autorizzazione con AWS Identity and Access Management (IAM). Per ulteriori informazioni, consulta Configurare l'autenticazione con IAM nella documentazione di MongoDB.	MongoDB DBA
Crea ruoli personalizzati.	(Facoltativo) Atlas supporta la creazione di ruoli personalizzati se i privilegi utente del database Atlas integrato non coprono il set di privilegi desiderato.	MongoDB DBA

Operazione	Description	Competenze richieste
Configura il peering VPC.	(Facoltativo) Atlas supporta il peering VPC con altri AWS. VPCs	MongoDB DBA
Configura un endpoint. AWS PrivateLink	(Facoltativo) Puoi configura re endpoint privati utilizzando AWS . AWS PrivateLink	MongoDB DBA
Abilita l'autenticazione a due fattori.	(Facoltativo) Atlas supporta l'autenticazione a due fattori (2FA) per aiutare gli utenti a controllare l'accesso ai propri account Atlas.	MongoDB DBA
Configura l'autenticazione e l'autorizzazione degli utenti con LDAP.	(Facoltativo) Atlas supporta l'esecuzione dell'autenticazion e e dell'autorizzazione degli utenti con Lightweight Directory Access Protocol (LDAP).	MongoDB DBA
Configura l'accesso unificato. AWS	(Facoltativo) Alcune funzional ità di Atlas, tra cui Atlas Data Lake e la crittografia a riposo mediante la gestione delle chiavi dei clienti, utilizzano i ruoli IAM per l'autenticazione.	MongoDB DBA
Configura la crittografia a riposo utilizzando. AWS KMS	(Facoltativo) Atlas supporta l'utilizzo di AWS Key Management Service (AWS KMS) per crittografare i motori di archiviazione e i backup dei provider di cloud.	MongoDB DBA

Operazione	Description	Competenze richieste
Configura CSFLE.	(Facoltativo) Atlas supporta la crittografia a livello di campo (CSFLE) lato client, inclusa la crittografia automatica dei campi.	MongoDB DBA

Migrazione dei dati

Operazione	Description	Competenze richieste
Avvia la replica di destinazione impostata in MongoDB Atlas.	Avvia la replica di destinazione impostata in MongoDB Atlas. In Atlas Live Migration Service, scegli Sono pronto per la migrazione.	MongoDB DBA
Stabilisci la connessione AWS Glue con MongoDB Atlas.	Usa an Crawler di AWS Glue per connetterti AWS Glue con MongoDB Atlas (database di destinazione). Questo passaggio aiuta a preparare l'ambiente di destinazione per la migrazione. Per ulteriori informazioni, consulta la documentazione relativa ad AWS Glue.	MongoDB DBA
Stabilisci la connessione AWS Glue con il database o il flusso di origine.	Questo aiuta a preparare l'ambiente di destinazione per la migrazione.	MongoDB DBA
Imposta la trasformazione dei dati.	Configura la logica di trasformazione per migrare i dati dallo schema strutturato	MongoDB DBA

Operazione	Description	Competenze richieste
	legacy allo schema flessibile di MongoDB.	
Migrare i dati.	Pianifica la migrazione in AWS Glue Studio.	MongoDB DBA

Configura l'integrazione operativa

Operazione	Description	Competenze richieste
Connect al cluster.	Connect al cluster MongoDB Atlas.	Sviluppatore di app
Interagisci con i dati.	Interagisci con i dati del cluster.	Sviluppatore di app
Monitora i cluster.	Monitora i tuoi cluster MongoDB Atlas.	MongoDB DBA
Backup e ripristino dei dati.	Eseguire il backup e il ripristino dei dati del cluster.	MongoDB DBA

Risoluzione dei problemi

Problema	Soluzione
Se riscontri problemi	Vedi Risoluzione dei problemi nel repository MongoDB CloudFormation Atlas Resources.

Risorse correlate

Tutti i seguenti collegamenti, se non diversamente specificato, portano alle pagine Web della documentazione di MongoDB.

Guida alla migrazione

- Migrazione a MongoDB Atlas AWS on (Guida prescrittiva)AWS

Scoperta e valutazione

- Memoria
- Esempio di dimensionamento con set di dati di esempio Atlas
- Esempio di dimensionamento per applicazioni mobili
- Traffico di rete
- Scalabilità automatica del cluster
- Modello di dimensionamento Atlas

Configurazione della sicurezza e della conformità

- Configurazione delle voci dell'elenco di accesso IP
- Configurazione degli utenti del database
- Configura l'accesso all'interfaccia utente Atlas
- Configura i ruoli personalizzati del database
- Configurazione degli utenti del database
- Configurare una connessione peering di rete
- Scopri di più sugli endpoint privati in Atlas
- Gestisci le tue opzioni di autenticazione a più fattori
- Configura l'autenticazione e l'autorizzazione degli utenti con LDAP
- Atlas Data Lake
- Crittografia a riposo con Customer Key Management
- Metodi per assumere un ruolo (documentazione IAM)
- Crittografia a livello di campo lato client
- Crittografia automatica
- Controlli di sicurezza MongoDB Atlas
- Centro fiducia MongoDB
- Configura le funzionalità di sicurezza per i cluster

Configurazione di un nuovo ambiente MongoDB Atlas su AWS

- Provider di cloud e regioni
- Gestisci i cluster globali
- Seleziona il livello del cluster
- Configura impostazioni aggiuntive
- Inizia con Atlas
- Configura l'accesso all'interfaccia utente Atlas

Migrazione dei dati

- Migrazione o importazione dei dati

Cluster di monitoraggio

- Monitora i tuoi cluster

Integrazione delle operazioni

- Connect a un cluster
- Interagisci con i tuoi dati
- Monitora i tuoi cluster
- Backup, ripristino e archiviazione dei dati

GitHub archivio

- Trasmetti dati a MongoDB Atlas utilizzando AWS Glue

Ruoli di transizione per un' PeopleSoft applicazione Oracle su Amazon RDS Custom for Oracle

sampath kathirvel, Amazon Web Services

Riepilogo

Per eseguire la soluzione Oracle PeopleSoft Enterprise Resource Planning (ERP) su Amazon Web Services (AWS), puoi utilizzare Amazon Relational Database Service (Amazon RDS) o Amazon RDS Custom per Oracle, che supporta applicazioni legacy, personalizzate e in pacchetti che richiedono l'accesso al sistema operativo (OS) e all'ambiente di database sottostanti. Per i fattori chiave da considerare quando si pianifica una migrazione, consulta le strategie di migrazione del database Oracle in AWS Prescriptive Guidance.

Questo modello si concentra sui passaggi per eseguire uno switchover di Oracle Data Guard, o transizione di ruolo, per un database di PeopleSoft applicazioni in esecuzione su Amazon RDS Custom come database primario con un database di replica di lettura. Il modello include i passaggi per configurare il failover ad avvio rapido (FSFO). Durante questo processo, i database nella configurazione di Oracle Data Guard continuano a funzionare nei loro nuovi ruoli. I casi d'uso tipici dello switchover di Oracle Data Guard sono le esercitazioni di disaster recovery (DR), le attività di manutenzione programmata sui database e le patch periodiche Standby-First Patch Apply. Per ulteriori informazioni, consulta il post del blog Ridurre i tempi di inattività delle patch del database in Amazon RDS Custom.

Prerequisiti e limitazioni

Prerequisiti

- Completamento dell'operazione Add HA to Oracle PeopleSoft on Amazon RDS Custom utilizzando un modello di replica di lettura.

Limitazioni

- Limitazioni e configurazioni non supportate per RDS Custom for Oracle
- Limitazioni associate alle repliche di lettura di Amazon RDS Custom for Oracle

Versioni del prodotto

- Per le versioni del database Oracle supportate da Amazon RDS Custom, consulta RDS Custom for Oracle.
- Per le classi di istanze di Oracle Database supportate da Amazon RDS Custom, consulta Supporto delle classi di istanze DB per RDS Custom for Oracle.

Architecture

Stack tecnologico

- Amazon RDS Custom per Oracle

Architettura Target

Il diagramma seguente mostra un'istanza DB personalizzata di Amazon RDS e una replica di lettura Amazon RDS Custom. Oracle Data Guard fornisce la transizione dei ruoli durante il failover per DR.

Per un'architettura rappresentativa che utilizza Oracle PeopleSoft su AWS, consulta Configurare un' PeopleSoft architettura ad alta disponibilità su AWS.

Tools (Strumenti)

Servizi AWS

- Amazon RDS Custom for Oracle è un servizio di database gestito per applicazioni legacy, personalizzate e confezionate che richiedono l'accesso al sistema operativo e all'ambiente di database sottostanti.
- AWS Secrets Manager ti aiuta a sostituire le credenziali codificate nel codice, comprese le password, con una chiamata API a Secrets Manager per recuperare il segreto a livello di codice. In questo modello, si recuperano le password degli utenti del database da Secrets Manager per `RDS_DATAGUARD` con il nome segreto. `do-not-delete-rds-custom-+<<RDS Resource ID>>+-dg`

Altri servizi

- Oracle Data Guard ti aiuta a creare, mantenere, gestire e monitorare i database in standby. Questo modello utilizza Oracle Data Guard Maximum Performance per i ruoli di transizione (Oracle Data Guard switchover).

Best practice

Per l'implementazione in produzione, consigliamo di avviare l'istanza observer in una terza zona di disponibilità, separata dai nodi di replica primari e di lettura.

Epiche

Avvia la transizione dei ruoli

Operazione	Description	Competenze richieste
Sospendi l'automazione del database sia per il database primario che per la replica.	Sebbene il framework di automazione RDS Custom non interferisca con il processo di transizione dei ruoli, è buona norma sospendere l'automazione durante il passaggio a Oracle Data Guard. Per sospendere e riprender e l'automazione del database RDS Custom, segui le istruzioni riportate in Sospensione e ripresa dell'automazione RDS Custom.	Amministratore cloud, DBA
Verifica lo stato di Oracle Data Guard.	Per verificare lo stato di Oracle Data Guard, accedi al database principale. Questo modello include il codice per l'utilizzo di un database contenitore multitenant (CDB) o un'istanza non CDB. Non CDB	DBA

Operazione	Description	Competenze richieste

	`-bash-4.2$ dgmgrl RDS_DATAGUARD@RDS_CUSTOM_ORCL_A`
	`DGMGRL for Linux: Release 19.0.0.0.0 - Production on Mon Nov 28 20:55:50 2022`
	`Version 19.10.0.0.0`
	`Copyright (c) 1982, 2019, Oracle and/or its affiliates. All rights reserved.`
	`Welcome to DGMGRL, type "help" for information.`
	`Password:`
	`Connected to "ORCL_A"`
	`Connected as SYSDG.`
	`DGMGRL> show configuration`
	`Configuration - rds_dg`
	`Protection Mode: MaxAvailability`
	`Members:`
	`orcl_a - Primary database`
	`orcl_d - Physical standby database`
	`Fast-Start Failover: Disabled`
	`Configuration Status:`
	`SUCCESS (status updated 59 seconds ago)`
	`DGMGRL>`
	
	CDB
	
	`CDB-bash-4.2$ dgmgrl C##RDS_DATAGUARD@RDS_CUSTOM_RDSCDB_A`	

Operazione	Description	Competenze richieste
	DGMGRL for Linux: Release 19.0.0.0.0 - Production on Wed Jan 18 06:13:07 2023 Version 19.16.0.0.0 Copyright (c) 1982, 2019, Oracle and/or its affiliates. All rights reserved. Welcome to DGMGRL, type "help" for informati on. Password: Connected to "RDSCDB_A " Connected as SYSDG. DGMGRL> show configura tion Configuration - rds_dg Protection Mode: MaxAvailability Members: rdscdb_a - Primary database rdscdb_b - Physical standby database Fast-Start Failover: Disabled Configuration Status: SUCCESS (status updated 52 seconds ago) DGMGRL>	

Operazione	Description	Competenze richieste
Verifica il ruolo dell'istanza.	Apri la Console di gestione AWS e accedi alla console Amazon RDS. Nella sezione Replica del database, nella scheda Connettività e sicurezza, verifica il ruolo dell'istanza per il primario e la replica. Il ruolo principale deve corrispondere al database primario di Oracle Data Guard e il ruolo di replica deve corrispondere al database di standby fisico di Oracle Data Guard.	Amministratore cloud, DBA

Operazione	Description	Competenze richieste

Esegui il passaggio.	Per eseguire lo switchover, connettiti a DGMGRL dal nodo principale.
	
	Non CDB
	
	`DGMGRL> switchover to orcl_d;`
	`Performing switchover NOW, please wait...`
	`Operation requires a connection to database "orcl_d"`
	`Connecting ...`
	`Connected to "ORCL_D"`
	`Connected as SYSDG.`
	`New primary database "orcl_d" is opening...`
	`Operation requires start up of instance "ORCL" on database "orcl_a"`
	`Starting instance "ORCL"...`
	`Connected to an idle instance.`
	`ORACLE instance started.`
	`Connected to "ORCL_A"`
	`Database mounted.`
	`Database opened.`
	`Connected to "ORCL_A"`
	`Switchover succeeded, new primary is "orcl_d"`
	`DGMGRL>`
	
	CDB
	
	`DGMGRL> switchover to rdscdb_b`
		DBA

Operazione	Description	Competenze richieste

	`Performing switchover NOW, please wait...`
	`New primary database "rdscdb_b" is opening...`
	`Operation requires start up of instance "RDSCDB" on database "rdscdb_a"`
	`Starting instance "RDSCDB"...`
	`Connected to an idle instance.`
	`ORACLE instance started.`
	`Connected to "RDSCDB_A"`
	`Database mounted.`
	`Database opened.`
	`Connected to "RDSCDB_A"`
	`Switchover succeeded, new primary is "rdscdb_b"`	

Operazione	Description	Competenze richieste

Verifica la connessione di Oracle Data Guard.	Dopo lo switchover, verifica la connessione Oracle Data Guard dal nodo principale a. `DGMGRL`
	
	Non CDB
	
	`DGMGRL> show configura tion; Configuration - rds_dg Protection Mode: MaxAvailability Members: orcl_d - Primary database orcl_a - Physical standby database Fast-Start Failover: Disabled Configuration Status: SUCCESS (status updated 60 seconds ago) DGMGRL>`
	
	`DGMGRL> show configura tion lag; Configuration - rds_dg Protection Mode: MaxAvailability Members: orcl_d - Primary database orcl_a - Physical standby database Transport Lag: 0 seconds (computed 0 seconds ago) Apply Lag: 0 seconds (computed 0 seconds ago)`
		DBA

Operazione	Description	Competenze richieste
	Fast-Start Failover: Disabled Configuration Status: SUCCESS (status updated 44 seconds ago) DGMGRL>	
	CDB	

	DGMGRL> show configura
	tion
	DGMGRL> show configura
	tion
	Configuration - rds_dg
	  Protection Mode:
	 MaxAvailability
	  Members:
	  rdscdb_b - Primary
	 database
	  rdscdb_a - Physical
	 standby database
	Fast-Start Failover:
	 Disabled
	Configuration Status:
	SUCCESS (status
	 updated 52 seconds ago)
	DGMGRL>
	
	DGMGRL> show configura
	tion lag
	Configuration - rds_dg
	  Protection Mode:
	 MaxAvailability
	  Members:
	  rdscdb_b - Primary
	 database
	  rdscdb_a - Physical
	 standby database
	  Transport
	 Lag: 0 seconds	

Operazione	Description	Competenze richieste
	(computed 0 seconds ago) Apply Lag: 0 seconds (computed 0 seconds ago) Fast-Start Failover: Disabled Configuration Status: SUCCESS (status updated 53 seconds ago) DGMGRL>	
Verifica il ruolo dell'istanza sulla console Amazon RDS.	Dopo aver eseguito il cambio di ruolo, la console Amazon RDS mostra i nuovi ruoli nella sezione Replica della scheda Connettività e sicurezza in Database. Potrebbero essere necessari alcuni minuti prima che lo stato di replica venga aggiornato da vuoto a Replicante.	DBA

Configurare FSFO

Operazione	Description	Competenze richieste
Reimposta lo switchover.	Reimposta lo switchover sul nodo principale.	DBA
Installa e avvia l'osservatore.	Un processo di osservazione è un componente `DGMGRL` client, in genere eseguito su una macchina diversa dai database primari e di standby.	DBA

Operazione	Description	Competenze richieste
	L'installazione di ORACLE HOME per l'observer può essere un'installazione di Oracle Client Administrator oppure è possibile installare Oracle Database Enterprise Edition o Personal Edition. Per ulteriori informazioni sull'installazione dell'observer per la versione del database, vedere Installazione e avvio dell'Observer. Per configurare l'alta disponibilità per il processo di osservazione, potresti voler fare quanto segue: • Abilita il ripristino automatico dell'EC2 istanza per l'EC2 istanza su cui è in esecuzione l'observer. È necessario automatizzare il processo di avvio dell'osservatore come parte dell'avvio del sistema operativo. • Implementa un osservatore nell' EC2 istanza e configura un gruppo Amazon EC2 Auto Scaling di dimensione uno (1). In caso di errore dell' EC2 istanza, il gruppo di ridimensionamento automatico avvia automaticamente un'altra istanza. EC2	

Operazione	Description	Competenze richieste
	Per Oracle 12c Release 2 e versioni successive, puoi implementare fino a tre osservatori. Un osservatore è l'osservatore principale e gli altri sono osservatori di riserva. Quando l'osservatore principale fallisce, uno degli osservatori di riserva assume il ruolo principale.	

Operazione	Description	Competenze richieste

Connect a DGMGRL dall'host dell'osservatore.	L'host dell'osservatore è configurato con `tnsnames.ora` voci per la connettività del database primario e di standby. È possibile abilitare FSFO con la modalità di protezione delle massime prestazioni purché la perdita di dati rientri nella FastStart FailoverLagLimitconfigurazione (valore in secondi). Tuttavia, è necessario utilizzare la modalità di protezione della massima disponibilità per ottenere una perdita di dati pari a zero (RPO=0).
	
	Non CDB
	
	`DGMGRL> show configuration;`
	`Configuration - rds_dg`
	`Protection Mode: MaxAvailability`
	`Members:`
	`orcl_a - Primary database`
	`orcl_d - Physical standby database`
	`Fast-Start Failover: Disabled`
	`Configuration Status:`
	`SUCCESS (status updated 58 seconds ago)`
	`DGMGRL> show configuration lag`
	`Configuration - rds_dg`
		DBA

Operazione	Description	Competenze richieste

```
Protection Mode:
 MaxAvailability
Members:
orcl_a - Primary
 database
orcl_d - Physical
 standby database
Transport Lag: 0
 seconds (computed 1
 second ago)
Apply Lag: 0 seconds
 (computed 1 second ago)
Fast-Start Failover:
 Disabled
Configuration Status:
SUCCESS (status updated
 5 seconds ago)
DGMGRL>
```

CDB

```
-bash-4.2$ dgmgrl
 C##RDS_DATAGUARD@R
DS_CUSTOM_RDSCDB_A
DGMGRL for Linux:
 Release 19.0.0.0.0 -
 Production on Wed Jan
 18 06:55:09 2023
Version 19.16.0.0.0
Copyright (c) 1982,
 2019, Oracle and/or
 its affiliates.  All
 rights reserved.
Welcome to DGMGRL, type
 "help" for informati
on.
Password:
Connected to "RDSCDB_A
"
Connected as SYSDG.
```

Operazione	Description	Competenze richieste
	`DGMGRL> show configura tion Configuration - rds_dg Protection Mode: MaxAvailability Members: rdscdb_a - Primary database rdscdb_b - Physical standby database Fast-Start Failover: Disabled Configuration Status: SUCCESS (status updated 18 seconds ago) DGMGRL>`	

Operazione	Description	Competenze richieste

Modificate il database di standby in modo che diventi la destinazione del failover.	Connect dal nodo primario o dal nodo osservatore a un database in standby. (Sebbene la configurazione possa avere più database in standby, è necessario connettersi a uno solo per volta.)
	
	Non CDB
	
	`DGMGRL> edit database orcl_a set property FastStartFailoverTarget='orcl_d';`
	`Property "faststartfailovertarget" updated`
	`DGMGRL> edit database orcl_d set property FastStartFailoverTarget='orcl_a';`
	`Property "faststartfailovertarget" updated`
	`DGMGRL> show database orcl_a FastStartFailoverTarget;`
	`FastStartFailoverTarget = 'orcl_d'`
	`DGMGRL> show database orcl_d FastStartFailoverTarget;`
	`FastStartFailoverTarget = 'orcl_a'`
	`DGMGRL>`
	
	CDB
		DBA

Operazione	Description	Competenze richieste

	DGMGRL> edit database
	 orcl_a set property
	 FastStartFailoverT
	arget='rdscdb_b';
	Object "orcl_a" was not
	 found
	DGMGRL> edit database
	 rdscdb_a set property
	 FastStartFailoverT
	arget='rdscdb_b';
	Property "faststar
	tfailovertarget"
	 updated
	DGMGRL> edit database
	 rdscdb_b set property
	 FastStartFailoverT
	arget='rdscdb_a';
	Property "faststar
	tfailovertarget"
	 updated
	DGMGRL> show database
	 rdscdb_a FastStart
	FailoverTarget;
	  FastStartFailoverT
	arget = 'rdscdb_b'
	DGMGRL> show database
	 rdscdb_b FastStart
	FailoverTarget;
	  FastStartFailoverT
	arget = 'rdscdb_a'
	DGMGRL>	

Operazione	Description	Competenze richieste

Configurare FastStart FailoverThreshold per la connessione a DGMGRL.	Il valore predefinito è 30 secondi in Oracle 19c e il valore minimo è 6 secondi. Un valore inferiore può potenzialmente ridurre il Recovery Time Objective (RTO) durante il failover. Un valore più alto aiuta a ridurre la possibilità di errori transitori di failover non necessari sul database primario.
	
	Il framework di automazione RDS Custom for Oracle monitora lo stato del database ed esegue azioni correttive ogni pochi secondi. Pertanto, si consiglia di impostare un valore superiore FastStart FailoverThreshold a 10 secondi. L'esempio seguente configura il valore di soglia a 35 secondi.
	
	Non CBD o CDB
	
	`DGMGRL> edit configuration set property FastStartFailoverThreshold=35;`
	`Property "faststartfailoverthreshold" updated`
	`DGMGRL> show configuration FastStart FailoverThreshold;`
		DBA

Operazione	Description	Competenze richieste
	`FastStartFailover Threshold = '35' DGMGRL>`	

Operazione	Description	Competenze richieste

Abilita FSFO connettendoti a DGMGRL dal nodo primario o osservatore.	Se il database non ha Flashback Database abilitato, viene visualizzato il messaggio di avviso. `ORA-16827` Il database flashback opzionale aiuta a ripristinare automaticamente i database primari guasti in un momento precedente al failover se la proprietà di FastStartFailoverAutoReinstateconfigurazione è impostata su TRUE (che è l'impostazione predefinita).
	
	Non CDB
	
	`DGMGRL> enable fast_start failover;`
	`Warning: ORA-16827: Flashback Database is disabled`
	`Enabled in Zero Data Loss Mode.`
	`DGMGRL>`
	`DGMGRL> show configuration`
	`Configuration - rds_dg`
	`Protection Mode: MaxAvailability`
	`Members:`
	`orcl_a - Primary database`
	`Warning: ORA-16819: fast-start failover observer not started`
	`orcl_d - (*) Physical standby database`
		DBA

Operazione	Description	Competenze richieste
	Warning: ORA-16819: fast-start failover observer not started Fast-Start Failover: Enabled in Zero Data Loss Mode Configuration Status: WARNING (status updated 29 seconds ago) DGMGRL> CDB DGMGRL> enable fast_star t failover; Warning: ORA-16827: Flashback Database is disabled Enabled in Zero Data Loss Mode. DGMGRL> show configura tion; Configuration - rds_dg Protection Mode: MaxAvailability Members: rdscdb_a - Primary database Warning: ORA-16819 : fast-start failover observer not started rdscdb_b - (*) Physical standby database Fast-Start Failover: Enabled in Zero Data Loss Mode Configuration Status: WARNING (status updated 11 seconds ago)	

Operazione	Description	Competenze richieste
	`DGMGRL>`	

Operazione	Description	Competenze richieste

Avvia l'osservatore per il monitoraggio FSFO e verifica lo stato.	È possibile avviare l'osservatore prima o dopo aver abilitato FSFO. Se FSFO è già abilitato, l'osservatore inizia immediatamente a monitorare lo stato e le connessioni ai database di standby primari e di destinazione. Se FSFO non è abilitato, l'osservatore inizia il monitoraggio solo dopo l'attivazione di FSFO.
	
	All'avvio dell'osservatore, la configurazione del DB principale verrà visualizzata senza messaggi di errore, come evidenziato dal comando precedente. `show configuration`
	
	Non CDB
	
	`DGMGRL> start observer;`
	`[W000 2022-12-01T06:16:51.271+00:00] FSFO target standby is orcl_d`
	`Observer 'ip-10-0-1-89' started`
	`[W000 2022-12-01T06:16:51.352+00:00] Observer trace level is set to USER`
	
	`DGMGRL> show configuration`
	`Configuration - rds_dg`
		DBA

Operazione	Description	Competenze richieste
	Protection Mode: MaxAvailability Members: orcl_a - Primary database orcl_d - (*) Physical standby database Fast-Start Failover: Enabled in Zero Data Loss Mode Configuration Status: SUCCESS (status updated 56 seconds ago) DGMGRL> DGMGRL> show observer Configuration - rds_dg Primary: orcl_a Active Target: orcl_d Observer "ip-10-0- 1-89" - Master Host Name: ip-10-0-1 -89 Last Ping to Primary: 1 second ago Last Ping to Target: 1 second ago DGMGRL> CDB DGMGRL> start observer; Succeeded in opening the observer file "/home/oracle/fsfo _ip-10-0-1-56.dat". [W000 2023-01-1 8T07:31:32.589+00:00] FSFO target standby is rdscdb_b	

Operazione	Description	Competenze richieste

	Observer 'ip-10-0-1-56' started
	The observer log file is '/home/oracle/observer_ip-10-0-1-56.log'.
	
	DGMGRL> show configuration
	Configuration - rds_dg
	  Protection Mode: MaxAvailability
	  Members:
	  rdscdb_a - Primary database
	    rdscdb_b - (*) Physical standby database
	Fast-Start Failover: Enabled in Zero Data Loss Mode
	Configuration Status:
	SUCCESS (status updated 12 seconds ago)
	DGMGRL>
	
	DGMGRL> show observer;
	Configuration - rds_dg
	  Primary: rdscdb_a
	  Active Target: rdscdb_b
	Observer "ip-10-0-1-56" - Master
	  Host Name: ip-10-0-1-56
	  Last Ping to Primary: 1 second ago
	  Last Ping to Target: 2 seconds ago
	DGMGRL>	

Operazione	Description	Competenze richieste
Verifica il failover.	In questo scenario, è possibile eseguire un test di failover arrestando manualmente l'istanza principale. EC2 Prima di arrestare l' EC2 istanza, utilizzate il `tail` comando per monitorare il file di registro dell'osservatore in base alla configurazione. Utilizzato `DGMGRL` per accedere al database in standby `orcl_d` con l'utente `RDS_DATAGUARD` e controllare lo stato di Oracle Data Guard. Dovrebbe mostrare che si `orcl_d` tratta del nuovo database primario. **Note** In questo scenario di test di failover, `orcl_d` è il database non CDB. Prima del failover, il database flashback era abilitato. `orcl_a` Dopo che il precedente database primario è tornato online e ha iniziato a funzionare, l'osservatore `MOUNT` lo ripristina in un nuovo database di standby. Il database ripristinato funge da destinazione FSFO per il nuovo database	DBA

Operazione	Description	Competenze richieste

	primario. È possibile verificare i dettagli nei log degli osservatori.
	
	`DGMGRL> show configuration`
	`Configuration - rds_dg`
	`Protection Mode: MaxAvailability`
	`Members:`
	`orcl_d - Primary database`
	`Warning: ORA-16824: multiple warnings, including fast-start failover-related warnings, detected for the database`
	`orcl_a - (*) Physical standby database (disabled)`
	`ORA-16661: the standby database needs to be reinstated`
	`Fast-Start Failover: Enabled in Zero Data Loss Mode`
	`Configuration Status: WARNING (status updated 25 seconds ago)`
	`DGMGRL>`
	
	Di seguito viene mostrato un esempio di output in. `observer.log`
	
	`$ tail -f /tmp/observer.log`	

Operazione	Description	Competenze richieste
	Unable to connect to database using rds_custom_orcl_a [W000 2023-01-18T07:50:32.589+00:00] Primary database cannot be reached. [W000 2023-01-18T07:50:32.589+00:00] Fast-Start Failover threshold has expired. [W000 2023-01-18T07:50:32.590+00:00] Try to connect to the standby. [W000 2023-01-18T07:50:32.590+00:00] Making a last connection attempt to primary database before proceeding with Fast-Start Failover. [W000 2023-01-18T07:50:32.591+00:00] Check if the standby is ready for failover. [S002 2023-01-18T07:50:32.591+00:00] Fast-Start Failover started... 2023-01-18T07:50:32.591+00:00 Initiating Fast-Start Failover to database "orcl_d"... [S002 2023-01-18T07:50:32.592+00:00] Initiating Fast-start Failover. Performing failover NOW, please wait...	

Operazione	Description	Competenze richieste
	Failover succeeded, new primary is "orcl_d" 2023-01-18T07:55:3 2.101+00:00 [S002 2023-01-1 8T07:55:32.591+00:00] Fast-Start Failover finished... [W000 2023-01-1 8T07:55:32.591+00:00] Failover succeeded. Restart pinging. [W000 2023-01-1 8T07:55:32.603+00:00] Primary database has changed to orcl_d. [W000 2023-01-1 8T07:55:33.618+00:00] Try to connect to the primary. [W000 2023-01-1 8T07:55:33.622+00: 00] Try to connect to the primary rds_custo m_orcl_d. [W000 2023-01-1 8T07:55:33.634+00: 00] The standby orcl_a needs to be reinstated [W000 2023-01-1 8T07:55:33.654+00:00] Try to connect to the new standby orcl_a. [W000 2023-01-1 8T07:55:33.654+00: 00] Connection to the primary restored! [W000 2023-01-1 8T07:55:35.654+00: 00] Disconnecting from database rds_custo m_orcl_d.	

Operazione	Description	Competenze richieste
	`[W000 2023-01-18T07:55:57.701+00:00] Try to connect to the new standby orcl_a. ORA-12170: TNS:Connect timeout occurred`	

Configura la connettività tra l'applicazione Oracle Peoplesoft e il database

Operazione	Description	Competenze richieste
Creare e avviare il servizio nel database principale.	È possibile evitare modifiche alla configurazione dell'applicazione durante una transizione di ruolo utilizzando una voce TNS che contiene sia gli endpoint del database primario che quelli di standby nella configurazione. È possibile definire due servizi di database basati sui ruoli per supportare entrambi i carichi di lavoro e quelli di sola lettura. read/write Nell'esempio seguente, `orcl_rw` è il read/write servizio attivo nel database primario. `orcl_ro`è il servizio di sola lettura ed è attivo nel database di standby che è stato aperto in modalità di sola lettura. `SQL> select name,open_mode from v$database; NAME OPEN_MODE`	DBA

Operazione	Description	Competenze richieste
	`--------- --------- ----------- ORCL READ WRITE SQL> exec dbms_serv ice.create_service ('orcl_rw','orcl_r w'); PL/SQL procedure successfully completed . SQL> exec dbms_serv ice.create_service ('orcl_ro','orcl_r o'); PL/SQL procedure successfully completed . SQL> exec dbms_serv ice.start_service('orcl_rw'); PL/SQL procedure successfully completed . SQL>`	

Operazione	Description	Competenze richieste
Avvia il servizio nel database di standby.	Per avviare il servizio nel database di standby di sola lettura, utilizzare il codice seguente. `SQL> select name,open _mode from v$database; NAME OPEN_MODE --------- ---------- ----------- ORCL READ ONLY WITH APPLY SQL> exec dbms_serv ice.start_service( 'orcl_ro'); PL/SQL procedure successfully completed . SQL>`	DBA

Operazione	Description	Competenze richieste
Automatizza l'avvio del servizio al riavvio del DB primario.	Per avviare automaticamente il servizio nel database primario al riavvio, usa il codice seguente. `SQL> CREATE OR REPLACE TRIGGER TrgDgServices after startup on database DECLARE db_role VARCHAR(30); db_open_mode VARCHAR(30); BEGIN SELECT DATABASE_ROLE, OPEN_MODE INTO db_role, db_open_mode FROM V$DATABASE; IF db_role = 'PRIMARY' THEN DBMS_SERV 2 ICE.START_SERVICE('orcl_rw'); END IF; IF db_role = 'PHYSICAL STANDBY' AND db_open_mode LIKE 'READ ONLY%' THEN DBMS_SERVICE.START_SERVICE('orcl_ro'); END IF; END; / Trigger created. SQL>`	DBA

Operazione	Description	Competenze richieste
Configura una connessione tra i database read/write e quelli di sola lettura.	È possibile utilizzare il seguente esempio di configurazione dell'applicazione per la connessione in sola lettura. read/write `ORCL_RW = (DESCRIPTION = (CONNECT_TIMEOUT= 120)(RETRY_COUNT=20)(RETRY_DELAY=3)(TRANSPORT_CONNECT_TIMEOUT=3) (ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCP)(HOST=devpsftdb.******.us-west-2.rds.amazonaws.com) (PORT=1521)) (ADDRESS = (PROTOCOL = TCP)(HOST=psftread.******.us-west-2.rds.amazonaws.com) (PORT=1521))) (CONNECT_DATA=(SERVICE_NAME = orcl_rw))) ORCL_RO = (DESCRIPTION = (CONNECT_TIMEOUT= 120)(RETRY_COUNT=20)(RETRY_DELAY=3)(TRANSPORT_CONNECT_TIMEOUT=3) (ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCP)(HOST=devpsftdb.******.us-west-2`	DBA

Operazione	Description	Competenze richieste
	`.rds.amazonaws.com)(PORT=1521))(ADDRESS = (PROTOCOL = TCP)(HOST=psftread.******.us-west-2.rds.amazonaws.com)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME = orcl_ro))`	

Risorse correlate

- Abilitare l'alta disponibilità con Data Guard su Amazon RDS Custom for Oracle (AWS Technical Guide)
- Configurazione di Amazon RDS come PeopleSoft database Oracle (white paper AWS)
- Guida Oracle Data Guard Broker (documentazione di riferimento Oracle)
- Concetti e amministrazione di Data Guard (documentazione di riferimento Oracle)
- Requisiti di configurazione FAN e FCF specifici per Oracle Data Guard (documentazione di riferimento Oracle)

Scarica i dati da un cluster Amazon Redshift tra più account su Amazon S3

Andrew Kamel, Amazon Web Services

Riepilogo

Quando si testano le applicazioni, è utile disporre dei dati di produzione nell'ambiente di test. L'utilizzo dei dati di produzione può darvi una valutazione più accurata dell'applicazione che state sviluppando.

Questo modello estrae i dati da un cluster Amazon Redshift in un ambiente di produzione in un bucket Amazon Simple Storage Service (Amazon S3) in un ambiente di sviluppo su Amazon Web Services ().AWS

Il modello illustra la configurazione degli account DEV e PROD, inclusi i seguenti:

- Risorse obbligatorie
- AWS Identity and Access Management ruoli (IAM)
- Modifiche di rete a sottoreti, gruppi di sicurezza e cloud privato virtuale (VPC) per supportare la connessione Amazon Redshift
- Una AWS Lambda funzione di esempio con un runtime Python per testare l'architettura

Per concedere l'accesso al cluster Amazon Redshift, il pattern utilizza il pattern Gestione dei segreti AWS per memorizzare le credenziali pertinenti. Il vantaggio è avere tutte le informazioni necessarie per connettersi direttamente al cluster Amazon Redshift senza dover sapere dove si trova il cluster Amazon Redshift. Inoltre, puoi monitorare l'uso del segreto.

Il segreto salvato in Secrets Manager include l'host del cluster Amazon Redshift, il nome del database, la porta e le credenziali pertinenti.

Per informazioni sulle considerazioni sulla sicurezza quando si utilizza questo modello, consulta la sezione Best practice.

Prerequisiti e limitazioni

Prerequisiti

- Un cluster Amazon Redshift in esecuzione nell'account PROD
- Un bucket S3 creato nell'account DEV
- Peering VPC tra gli account DEV e PROD, con tabelle di routing adattate di conseguenza

- Nomi host DNS e risoluzione DNS abilitati per entrambi i sistemi peer-to-peer VPCs

Limitazioni

- A seconda della quantità di dati da interrogare, la funzione Lambda potrebbe scadere.

  Se l'esecuzione richiede più tempo del timeout Lambda massimo (15 minuti), utilizza un approccio asincrono per il codice Lambda. L'esempio di codice per questo pattern utilizza la libreria psycopg2 per Python, che attualmente non supporta l'elaborazione asincrona.
- Alcuni non sono disponibili in tutti. Servizi AWS Regioni AWS Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Architecture

Il diagramma seguente mostra l'architettura di destinazione, con gli account DEV e PROD.

Il diagramma mostra il flusso di lavoro seguente:

1. La funzione Lambda nell'account DEV presuppone il ruolo IAM necessario per accedere alle credenziali Amazon Redshift in Secrets Manager nell'account PROD.

   La funzione Lambda recupera quindi il segreto del cluster Amazon Redshift.
2. La funzione Lambda nell'account DEV utilizza le informazioni per connettersi al cluster Amazon Redshift nell'account PROD tramite il peered. VPCs

   La funzione Lambda invia quindi un comando unload per interrogare il cluster Amazon Redshift nell'account PROD.
3. Il cluster Amazon Redshift nell'account PROD assume il ruolo IAM pertinente per accedere al bucket S3 nell'account DEV.

   Il cluster Amazon Redshift scarica i dati richiesti nel bucket S3 nell'account DEV.

Interrogazione di dati da Amazon Redshift

Il diagramma seguente mostra i ruoli utilizzati per recuperare le credenziali di Amazon Redshift e connettersi al cluster Amazon Redshift. Il flusso di lavoro viene avviato dalla funzione Lambda.

Il diagramma mostra il flusso di lavoro seguente:

1. L'account `CrossAccount-SM-Read-Role` nell'account DEV presuppone quello `SM-Read-Role` nell'account PROD.
2. Il `SM-Read-Role` ruolo utilizza la policy allegata per recuperare il segreto da Secrets Manager.
3. Le credenziali vengono utilizzate per accedere al cluster Amazon Redshift.

Caricamento di dati su Amazon S3

Il diagramma seguente mostra il processo di lettura-scrittura tra account per l'estrazione dei dati e il loro caricamento su Amazon S3. Il flusso di lavoro viene avviato dalla funzione Lambda. Il pattern concatena i ruoli IAM in Amazon Redshift. Il comando unload proveniente dal cluster Amazon Redshift presuppone, e `CrossAccount-S3-Write-Role` quindi assume. `S3-Write-Role` Questo concatenamento di ruoli consente ad Amazon Redshift di accedere ad Amazon S3.

Il flusso di lavoro include i seguenti passaggi:

1. L'accesso `CrossAccount-SM-Read-Role` all'account DEV presuppone l'accesso `SM-Read-Role` all'account PROD.
2. `SM-Read-Role`recupera le credenziali di Amazon Redshift da Secrets Manager.
3. La funzione Lambda si connette al cluster Amazon Redshift e invia una query.
4. Il cluster Amazon Redshift presuppone il. `CrossAccount-S3-Write-Role`
5. `CrossAccount-S3-Write-Role`Si presuppone che sia presente `S3-Write-Role` nell'account DEV.
6. I risultati della query vengono scaricati nel bucket S3 nell'account DEV.

Tools (Strumenti)

Servizi AWS

- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.

- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Redshift è un servizio di data warehouse gestito su scala petabyte nel cloud AWS.
- Con Gestione dei segreti AWS puoi sostituire le credenziali nel codice,incluse le password, con una chiamata API a Secrets Manager in modo da recuperare il segreto a livello di codice.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Repository di codice

Il codice per questo pattern è disponibile nel repository GitHub unload-redshift-to-s3-python.

Best practice

Dichiarazione di non responsabilità in materia di sicurezza

Prima di implementare questa soluzione, prendete in considerazione i seguenti importanti consigli di sicurezza:

- Ricorda che collegare gli account di sviluppo e produzione può aumentare l'ambito e ridurre il livello di sicurezza complessivo. Consigliamo di implementare questa soluzione solo temporaneamente, estraendo la parte di dati richiesta e quindi distruggendo immediatamente le risorse distribuite. Per distruggere le risorse, è necessario eliminare la funzione Lambda, rimuovere tutti i ruoli e le policy IAM creati per questa soluzione e revocare qualsiasi accesso alla rete concesso tra gli account.
- Consulta i tuoi team di sicurezza e conformità prima di copiare qualsiasi dato dagli ambienti di produzione a quelli di sviluppo. Le informazioni di identificazione personale (PII), le informazioni sanitarie protette (PHI) e altri dati riservati o regolamentati in genere non devono essere copiati in questo modo. Copia solo le informazioni non riservate disponibili al pubblico (ad esempio, dati di stock pubblici provenienti dall'interfaccia di un negozio). Prendi in considerazione la tokenizzazione o l'anonimizzazione dei dati o la generazione di dati di test sintetici, invece di utilizzare i dati di produzione quando possibile. Uno dei principi di AWS sicurezza è tenere le persone lontane dai dati. In altre parole, gli sviluppatori non devono eseguire operazioni nell'account di produzione.
- Limita l'accesso alla funzione Lambda nell'account di sviluppo perché può leggere i dati dal cluster Amazon Redshift nell'ambiente di produzione.
- Per evitare di interrompere l'ambiente di produzione, implementa i seguenti consigli:
  - Utilizzate un account di sviluppo separato e dedicato per le attività di test e sviluppo.

  - Implementa rigorosi controlli di accesso alla rete e limita il traffico tra gli account solo a quanto necessario.
  - Monitora e verifica l'accesso all'ambiente di produzione e alle fonti di dati.
  - Implementa controlli di accesso con privilegi minimi per tutte le risorse e i servizi coinvolti.
  - Esamina e ruota regolarmente le credenziali, come i Gestione dei segreti AWS segreti e le chiavi di accesso ai ruoli IAM.
- Consulta la seguente documentazione sulla sicurezza per i servizi utilizzati in questo articolo:
  - AWS Lambda sicurezza
  - Sicurezza di Amazon Redshift
  - Sicurezza di Amazon S3
  - Gestione dei segreti AWS sicurezza
  - Best practice sulla sicurezza IAM

La sicurezza è una priorità assoluta quando si accede ai dati e alle risorse di produzione. Segui sempre le migliori pratiche, implementa controlli di accesso con privilegi minimi e rivedi e aggiorna regolarmente le tue misure di sicurezza.

Epiche

Interroga i dati da Amazon Redshift

Operazione	Description	Competenze richieste
Crea un segreto per il cluster Amazon Redshift.	Per creare il segreto per il cluster Amazon Redshift, procedi come segue: 1. Nell'account PROD, accedi a e apri Console di gestione AWS la console Secrets Manager all'indirizzo https://console.aws.amazon.com/secretsmanager/. 2. Scegli Memorizza un nuovo segreto.	DevOps ingegnere

Operazione	Description	Competenze richieste
	3. Seleziona Credenziali per il data warehouse Amazon Redshift. 4. Per Nome utente e password, inserisci i valori per l'istanza e conferma o scegli un valore per la chiave di crittografia. 5. Scegli il data warehouse di Amazon Redshift a cui accederà il tuo segreto. 6. Inserisci `Redshift-Creds-Secret` il nome segreto. 7. Completa i restanti passaggi di creazione con le scelte predefinite, quindi scegli Store. 8. Visualizza il tuo segreto e annota il valore ARN segreto generato per identificare il segreto.	

Operazione	Description	Competenze richieste
Crea un ruolo per accedere a Secrets Manager.	Per creare il ruolo, procedi come segue: 1. Nell'account PROD, apri la console IAM all'indirizzo https://console.aws.amazon.com/iam/. 2. Seleziona Policy. 3. Scegli Crea policy. 4. Scegli la scheda JSON, quindi inserisci una policy IAM come la seguente: `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [ "secretsmanager:GetResourcePolicy", "secretsmanager:GetSecretValue", "secretsmanager:DescribeSecret", "secretsmanager:ListSecretVersionIds" ], "Resource": [`	DevOps ingegnere

Operazione	Description	Competenze richieste

	<pre>"<Redshift-Creds-Secret-ARN>"
	            ]
	        },
	        {
	            "Effect": "Allow",
	            "Action": "secretsmanager:ListSecrets",
	            "Resource": "*"
	        }
	    ]
	}</pre>
	Sostituisci `Redshift-Creds-Secret-ARN` con l'Amazon Resource Name (ARN) del segreto Secrets Manager che contiene le informazioni e le credenziali per il cluster Amazon Redshift.	

Caricamento dei dati su Amazon S3

Operazione	Description	Competenze richieste
Crea un ruolo per accedere al bucket S3.	Per creare il ruolo per l'accesso al bucket S3, procedi come segue: 1. Nell'account DEV, apri la console IAM. 2. Seleziona Policy.	DevOps ingegnere

Operazione	Description	Competenze richieste
	3. Scegli Crea policy. 4. Scegli la scheda JSON, quindi inserisci una policy IAM come la seguente: `{ "Version": "2012-10-17", "Statement": [{ "Sid": "kmsstmt", "Effect": "Allow", "Action": ["kms:Decrypt", "kms:Encrypt", "kms:GenerateDataKey"], "Resource": ["<kms-key-arn>"] }, { "Sid": "s3stmt", "Effect": "Allow", "Action": ["s3:PutObject", "s3:Get*", "s3:List*"], "Resource": ["arn:aws:s3:::mybucket",`	

Operazione	Description	Competenze richieste

	<pre>"arn:aws:
	s3:::mybucket/*"
	]
	}
	]
	}</pre>
	Sostituiscilo `mybucket` con il nome del bucket S3 a cui desideri accedere. Inoltre, se il bucket S3 è crittografato, sostituiscilo `kms-key-arn` con l'ARN della chiave AWS Key Management Service (AWS KMS) utilizzata per crittografare il bucket S3. Altrimenti, non è necessaria la sezione della policy. AWS KMS
	5. Scegli Rivedi politica, inserisci `S3-Write-Policy` come nome della politica, quindi scegli Crea politica.
	6. Nel pannello di navigazione, seleziona Roles (Ruoli).
	7. Selezionare Create role (Crea ruolo).
	8. Per il ruolo di entità attendibile, scegli Politica di fiducia personalizzata.
	9. Scegli Avanti: Autorizzazioni, quindi seleziona la politica S3-Write-Policy che hai creato.	

Operazione	Description	Competenze richieste
	10. Immettete **`S3-Write-Role`** come nome del ruolo, quindi scegliete Crea ruolo.	

Operazione	Description	Competenze richieste
Crea il ruolo Amazon Redshift.	Per creare il ruolo Amazon Redshift, procedi come segue: 1. Nell'account PROD, apri la console IAM. 2. Seleziona Policy. 3. Scegli Crea policy. 4. Scegli la scheda JSON, quindi inserisci una policy IAM come la seguente: `{ "Version": "2012-10-17", "Statement": [ { "Sid": "CrossAccountPolicy", "Effect": "Allow", "Action": "sts:AssumeRole", "Resource": "S3-Write-Role-ARN" } ] }` Sostituisci `S3-Write-Role-ARN` con l'ARN dell'account `S3-Write-Role` DEV. 5. Scegli Review policy, inserisci `S3-Write-Role-Assume-Policy`	DevOps ingegnere

Operazione	Description	Competenze richieste
	come nome della policy, quindi scegli Crea policy. 6. Nel pannello di navigazione, scegli Ruoli e quindi Crea ruolo. 7. Scegli il servizio AWS come tipo di entità affidabile, quindi scegli Redshift, Redshift - Customizzabile. 8. Scegli Avanti: Autorizzazioni, quindi seleziona la policy S3-Write-Role-Assume-Policy che hai creato. 9. Immettete **`CrossAccount-S3-Write-Role`** come nome del ruolo, quindi scegliete Crea ruolo. 10. Associa il ruolo IAM al tuo cluster Amazon Redshift.	

Distribuire la funzione Lambda

Operazione	Description	Competenze richieste
Distribuire la funzione Lambda.	Per implementare una funzione Lambda nel VPC peerizzato, procedi come segue: 1. Apri la console Lambda all'indirizzo. https://console.aws.amazon.com/lambda/	DevOps ingegnere

Operazione	Description	Competenze richieste
	2. Scegli Funzioni. 3. Scegli Crea funzione. 4. In Informazioni di base, immettere un nome per la funzione in Nome funzione. 5. Per Runtime, scegli Python 3.8. 6. Espandi Cambia il ruolo di esecuzione predefinito, quindi procedi come segue: a. Scegli Usa un ruolo esistente. b. Per Ruolo esistente , seleziona il ruolo Lambda CrossAccount-RM-Read-Role creato in precedenza. 7. Espandi Impostazioni avanzate ed esegui le seguenti operazioni: a. Seleziona la casella di controllo Abilita VPC. b. Per VPC, seleziona il VPC peered nell'account DEV. c. Per Subnet, seleziona la sottorete privata. d. In Security groups (Gruppi di sicurezza), selezionare il gruppo di sicurezza predefinito. 8. Scegli Crea funzione.	

Operazione	Description	Competenze richieste
	9. Aggiungi la libreria psycopg2 come livello alla funzione Lambda. **Note** È possibile utilizzare un layer già distribuito dal repository psycopg2-lambda-layer. Assicurati di utilizzare l'URL in base al tuo runtime Regione AWS e a quello di Python.	

Testa l'architettura

Operazione	Description	Competenze richieste

Importa le risorse richieste.	Per importare le risorse richieste, esegui i seguenti comandi:
	
	`import ast`
	`import boto3`
	`import psycopg2`
	`import base64`
	`from botocore.exceptions import ClientError`
		Sviluppatore di app

Esegui la funzione di gestione Lambda.	La funzione Lambda utilizza AWS Security Token Service	Sviluppatore di app

Operazione	Description	Competenze richieste

	(AWS STS) per l'accesso tra account e la gestione temporanea delle credenziali. La funzione utilizza l'operazione AssumeRole API per assumere temporaneamente le autorizzazioni del ruolo IAM. `sm_read_role`
	
	Per eseguire la funzione Lambda, usa il seguente codice di esempio:
	
	`def lambda_handler(event, context):`
	`  sts_client = boto3.client('sts')`
	
	`  # Secrets Manager Configurations`
	`  secret_name = "redshift_creds"`
	`  sm_region = "eu-west-1"`
	`  sm_read_role = "arn:aws:iam::PROD_ACCOUNT_NUMBER:role/SM-Read-Role"`
	
	`  # S3 Bucket Configurations`
	`  s3_bucket_path = "s3://mybucket/"`
	`  s3_bucket_region = "eu-west-1"`
	`  s3_write_role = "arn:aws:iam::DEV_ACCOUNT_NUMBER:role/S3-Write-Role"`	

Operazione	Description	Competenze richieste

	`# Redshift Configurations`
	`sql_query = "select * from category"`
	`redshift_db = "dev"`
	`redshift_s3_write_role = "arn:aws:iam::PROD_ACCOUNT_NUMBER:role/CrossAccount-S3-Write-Role"`
	
	`chained_s3_write_role = "%s,%s" % (redshift_s3_write_role, s3_write_role)`
	
	`assumed_role_object = sts_client.assume_role(`
	`RoleArn=sm_read_role,`
	`RoleSessionName="CrossAccountRoleAssumption",`
	`ExternalId="YOUR_EXTERNAL_ID",`
	`)`
	`credentials = assumed_role_object['Credentials']`
	
	`secret_dict = ast.literal_eval(get_secret(credentials, secret_name, sm_region))`
	`execute_query(secret_dict, sql_query, s3_bucket_path, chained_s3_write_role, s3_bucket_region, redshift_db)`	

Operazione	Description	Competenze richieste
	`return { 'statusCode': 200 }`	

Operazione	Description	Competenze richieste

Scoprite il segreto.	Per ottenere il segreto di Amazon Redshift, usa il seguente codice di esempio:
	
	`def get_secret(credentials, secret_name, sm_region):`
	`  # Create a Secrets Manager client`
	`  session = boto3.session.Session()`
	`  sm_client = session.client(`
	`  service_name='secretsmanager',`
	`  aws_access_key_id=credentials['AccessKeyId'],`
	`  aws_secret_access_key=credentials['SecretAccessKey'],`
	`  aws_session_token=credentials['SessionToken'],`
	`  region_name=sm_region`
	`  )`
	
	`  try:`
	`  get_secret_value_response = sm_client.get_secret_value(`
	`  SecretId=secret_name`
	`  )`
	`  except ClientError as e:`
	`  print(e)`
	`  raise e`
		Sviluppatore di app

Operazione	Description	Competenze richieste

	`else:`
	`if 'SecretString' in get_secret_value_response:`
	`return get_secret_value_response['SecretString']`
	`else:`
	`return base64.b64decode(get_secret_value_response['SecretBinary'])`	

Operazione	Description	Competenze richieste

Esegui il comando unload.	Per scaricare i dati nel bucket S3, usa il seguente codice di esempio.
	
	`def execute_query(secret_dict, sql_query, s3_bucket_path, chained_s3_write_role, s3_bucket_region, redshift_db):`
	`  conn_string = "dbname='%s' port='%s' user='%s' password='%s' host='%s'" \`
	`  %(redshift_db,`
	`  secret_dict["port"],`
	`  secret_dict["username"],`
	`  secret_dict["password"],`
	`  secret_dict["host"])`
	`  con = psycopg2.connect(conn_string)`
	`  unload_command = "UNLOAD ('{}') TO '{}' IAM_ROLE '{}' DELIMITER '|' REGION '{}';" \`
	`  .format(sql_query,`
	`  s3_bucket_path + str(datetime.datetime.now()) + ".csv",`
		Sviluppatore di app

Operazione	Description	Competenze richieste

	``
	chained_s3_write_role,
	
	s3_bucket_region)
	
	# Opening a cursor and run query
	cur = con.cursor()
	cur.execute(unload_command)
	
	print(cur.fetchone())
	cur.close()
	con.close()
	``	

Eliminazione

Operazione	Description	Competenze richieste
Elimina la funzione Lambda.	Per evitare di incorrere in costi non pianificati, rimuovi le risorse e la connessione tra gli account DEV e PROD. Per rimuovere la funzione Lambda, procedi come segue: 1. Apri la AWS Lambda console all'indirizzo https://console.aws.amazon.com/lambda/. 2. Individua e seleziona la funzione Lambda che hai creato.	DevOps ingegnere

Operazione	Description	Competenze richieste
	3. Scegli Azioni, quindi Elimina. 4. Conferma l’eliminazione.	

Operazione	Description	Competenze richieste
Rimuovi i ruoli e le politiche IAM.	Rimuovi i ruoli e le policy IAM dagli account DEV e PROD. Nell'account DEV, procedi come segue: 1. Apri la console IAM. 2. Eliminate i seguenti ruoli: • `S3-Write-Role` • `CrossAccount-RM-Read-Role` (ruolo Lambda) 3. Eliminare le politiche associate: • `S3-Write-Policy` • La CrossAccount politica per l'assunzione dei ruoli degli account PROD Nell'account PROD, procedi come segue: 1. Apri la console IAM. 2. Eliminare i seguenti ruoli: • `SM-Read-Role` • `CrossAccount-S3-Write-Role` 3. Eliminare le politiche associate: • La CrossAccount politica di accesso a Secrets Manager	DevOps ingegnere

Operazione	Description	Competenze richieste
	• `S3-Write-Role-Assume-Policy`	
Eliminare il segreto in Secrets Manager.	Per eliminare il segreto, procedi come segue: 1. Nell'account PROD, apri la console Secrets Manager. 2. Individua e seleziona il segreto denominat o`Redshift-Creds-Secret` . 3. Scegli Actions (Operazioni) e Delete secret (Elimina segreto). 4. Conferma l’eliminazione.	DevOps ingegnere

Operazione	Description	Competenze richieste
Rimuovi le regole del peering e dei gruppi di sicurezza VPC.	Per rimuovere le regole del peering e dei gruppi di sicurezza VPC, procedi come segue: 1. Nell'account PROD, apri la EC2 console Amazon all'indirizzo https://console.aws.amazon.com/ec2/. 2. Passa a Gruppi di sicurezza. 3. Trova il gruppo di sicurezza utilizzato dal cluster Amazon Redshift. 4. Modifica le regole in entrata e rimuovi la regola che consente le connessioni dal VPC Lambda dell'account DEV. 5. Vai a Connessioni peering VPC ed elimina la connessione peering.	DevOps ingegnere

Operazione	Description	Competenze richieste
Rimuovi i dati dal bucket S3.	Per rimuovere i dati da Amazon S3, procedi come segue: 1. Nell'account DEV, apri la console Amazon S3 all'indirizzo. https://console.aws.amazon.com/s3/ 2. Individua il bucket che hai usato per l'archiviazione dei dati. 3. Elimina gli oggetti all'interno del bucket o elimina l'intero bucket se non è più necessario.	DevOps ingegnere
Pulisci AWS KMS le chiavi.	Se hai creato delle AWS KMS chiavi personalizzate per la crittografia, procedi come segue: 1. Apri la AWS KMS console all'indirizzo https://console.aws.amazon.com/kms/. 2. Individua tutte le chiavi create per questo modello. 3. Pianifica la cancellazione delle chiavi. (Esiste un periodo di attesa obbligatorio per l'eliminazione delle chiavi).	DevOps ingegnere

Operazione	Description	Competenze richieste
Controlla ed elimina i CloudWatch log di Amazon.	Per eliminare i CloudWatch log, procedi come segue: 1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/. 2. Verifica la presenza di eventuali gruppi di log creati dalla funzione Lambda o dal cluster Amazon Redshift. 3. Elimina questi gruppi di log se non sono più necessari.	DevOps ingegnere

Risorse correlate

- CloudWatch Documentazione Amazon
- Documentazione IAM
- Documentazione Lambda
- Documentazione Amazon Redshift
- Documentazione Amazon S3
- Gestione dei segreti AWS documentazione
- AWS principi di sicurezza

Informazioni aggiuntive

Dopo aver scaricato i dati da Amazon Redshift ad Amazon S3, puoi analizzarli utilizzando Amazon Athena.

Amazon Athena è un servizio di interrogazione per Big Data utile quando è necessario accedere a grandi volumi di dati. È possibile utilizzare Athena senza dover effettuare il provisioning di server o database. Athena supporta interrogazioni complesse ed è possibile eseguirla su oggetti diversi.

Come per la maggior parte Servizi AWS, il vantaggio principale dell'utilizzo di Athena è che offre una grande flessibilità nel modo in cui vengono eseguite le query senza complessità aggiuntiva. Quando usi Athena, puoi interrogare diversi tipi di dati, come CSV e JSON, in Amazon S3 senza modificare il tipo di dati. Puoi interrogare i dati da varie fonti, anche esterne. AWS Athena riduce la complessità perché non è necessario gestire i server. Athena legge i dati direttamente da Amazon S3 senza caricarli o modificarli prima di eseguire la query.

Modelli di migrazione del database per carico di lavoro

Argomenti

- IBM
- Microsoft
- N/D
- Open source
- Oracle
- SAP

IBM

- Esegui la migrazione di un database Db2 da Amazon a EC2 Aurora compatibile con MySQL utilizzando AWS DMS
- Esegui la migrazione di Db2 for LUW ad Amazon EC2 utilizzando la spedizione di log per ridurre i tempi di interruzione
- Esegui la migrazione di Db2 per LUW ad Amazon EC2 con disaster recovery ad alta disponibilità
- Esegui la migrazione da IBM Db2 su Amazon a EC2 Aurora PostgreSQL compatibile con AWS DMS e AWS SCT
- Esegui la migrazione da IBM WebSphere Application Server ad Apache Tomcat su Amazon EC2
- Trasmetti dati da IBM Db2, SAP, Sybase e altri database a MongoDB Atlas su AWS

Microsoft

- Accelera la scoperta e la migrazione dei carichi di lavoro Microsoft su AWS
- Accedi alle tabelle Microsoft SQL Server locali da Microsoft SQL Server su Amazon EC2 utilizzando server collegati
- Valuta le prestazioni delle query per la migrazione dei database SQL Server su MongoDB Atlas su AWS
- Automatizza le attività del database nell'edizione SQL Server Express in esecuzione su Amazon EC2 utilizzando Task AWS Lambda Scheduler
- Modifica le applicazioni Python e Perl per supportare la migrazione dei database da Microsoft SQL Server a Amazon Aurora PostgreSQL Compatible Edition
- Configurare il routing di sola lettura in un gruppo di disponibilità Always On in SQL Server su AWS
- Configurare l'autenticazione Windows per Amazon RDS for Microsoft SQL Server utilizzando AWS Managed Microsoft AD
- Crea CloudFormation modelli AWS per attività AWS DMS utilizzando Microsoft Excel e Python
- Distribuisci istanze di cluster di failover di SQL Server su Amazon e EC2 Amazon FSx utilizzando Terraform
- Esportazione di un database Microsoft SQL Server in Amazon S3 utilizzando AWS DMS
- Esporta tabelle Amazon RDS for SQL Server in un bucket S3 utilizzando AWS DMS
- SHA1 Implementa l'hashing per i dati PII durante la migrazione da SQL Server a PostgreSQL
- Acquisisci e migra istanze EC2 Windows in un account AWS Managed Services
- Configura un cluster di failover di Microsoft SQL Server su Amazon EC2 utilizzando FSx per Windows File Server
- Esegui la migrazione di una coda di messaggistica da Microsoft Azure Service Bus ad Amazon SQS
- Esegui la migrazione di un database Microsoft SQL Server da Amazon EC2 ad Amazon DocumentDB utilizzando AWS DMS
- Esegui la migrazione di un database Microsoft SQL Server su Aurora MySQL utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un'applicazione.NET da Microsoft Azure App Service ad AWS Elastic Beanstalk
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon EC2
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon RDS for SQL Server

- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon RDS for SQL Server utilizzando server collegati
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon RDS for SQL Server utilizzando metodi di backup e ripristino nativi
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon Redshift utilizzando AWS DMS
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon Redshift utilizzando gli agenti di estrazione dati AWS SCT
- Esegui la migrazione di un database Microsoft SQL Server locale a Microsoft SQL Server su Amazon EC2 con Linux
- Esegui la migrazione dei dati da Microsoft Azure Blob ad Amazon S3 utilizzando Rclone
- Esegui la migrazione di applicazioni ospitate da IIS su Amazon utilizzando appcmd.exe EC2
- Esegui la migrazione del gruppo di disponibilità Microsoft SQL Server Always On utilizzando AWS Application Migration Service
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon EC2 utilizzando Application Migration Service
- Migrare un database relazionale su MongoDB Atlas su AWS
- Esegui la migrazione di SQL Server su AWS utilizzando gruppi di disponibilità distribuiti
- Migrazione dei certificati SSL di Windows su un Application Load Balancer utilizzando ACM
- Rehosting dei carichi di lavoro locali nel cloud AWS: lista di controllo per la migrazione
- Risolvi gli errori di connessione dopo la migrazione di Microsoft SQL Server al cloud AWS
- Invia notifiche per un'istanza di database Amazon RDS for SQL Server utilizzando un server SMTP locale e Database Mail
- Configura una CI/CD pipeline per la migrazione del database utilizzando Terraform
- Configura un'infrastruttura Multi-AZ per SQL Server Always On FCI utilizzando Amazon FSx

N/D

- Creare un processo di approvazione per le richieste del firewall durante una migrazione di rehosting a AWS
- Crittografa un'istanza database Amazon RDS for PostgreSQL esistente
- Stima dei costi di storage per una tabella Amazon DynamoDB
- Implementa il disaster recovery tra regioni con AWS DMS e Amazon Aurora

Open source

- Genera automaticamente un modello PynamoDB e funzioni CRUD per Amazon DynamoDB utilizzando un'applicazione Python
- Connect utilizzando un tunnel SSH in pGAdmin
- Crea utenti e ruoli delle applicazioni in Aurora, compatibile con PostgreSQL
- Abilita connessioni crittografate per le istanze DB PostgreSQL in Amazon RDS
- Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL con e utilizzando e AWS SCT AWS DMS AWS CLI CloudFormation
- Esegui la migrazione di un database MariaDB locale su Amazon RDS for MariaDB utilizzando strumenti nativi
- Esegui la migrazione di un database MySQL locale su Amazon EC2
- Esegui la migrazione di un database MySQL locale su Amazon RDS for MySQL
- Esegui la migrazione di un database MySQL locale su Aurora MySQL
- Esegui la migrazione di un database PostgreSQL locale su Aurora PostgreSQL
- Esegui la migrazione di un database Couchbase Server su Amazon EC2
- Esegui la migrazione da IBM WebSphere Application Server ad Apache Tomcat su Amazon EC2 con Auto Scaling
- Esegui la migrazione da Oracle 8i o 9i ad Amazon RDS for Oracle utilizzando AWS DMS SharePlex
- Esegui la migrazione da PostgreSQL su Amazon RDS per PostgreSQL utilizzando EC2 pglogical
- Esegui la migrazione di applicazioni Java locali su AWS utilizzando AWS App2Container
- Esegui la migrazione dei database MySQL locali su Aurora MySQL utilizzando Percona, Amazon EFS e Amazon S3 XtraBackup
- Esegui la migrazione di tabelle esterne Oracle verso Amazon Aurora, compatibile con PostgreSQL
- Esegui la migrazione di funzioni e procedure Oracle con più di 100 argomenti a PostgreSQL
- Esegui la migrazione dei carichi di lavoro Redis su Redis Enterprise Cloud su AWS
- Monitora Amazon Aurora per le istanze senza crittografia
- Riavvia automaticamente l'AWS Replication Agent senza disabilitarlo SELinux dopo il riavvio di un server di origine RHEL
- Pianifica i lavori per Amazon RDS for PostgreSQL e Aurora PostgreSQL utilizzando Lambda e Secrets Manager

- Trasporta i database PostgreSQL tra due istanze DB Amazon RDS utilizzando pg_transport
- Scarica i dati da un cluster Amazon Redshift tra più account su Amazon S3

Oracle

- Aggiungi HA a Oracle PeopleSoft su Amazon RDS Custom utilizzando una replica di lettura
- Convertire le query JSON Oracle in SQL del database PostgreSQL
- Converti VARCHAR2 (1) il tipo di dati per Oracle in tipo di dati booleano per Amazon Aurora PostgreSQL
- Emula Oracle DR utilizzando un database globale Aurora compatibile con PostgreSQL
- Emula gli array PL/SQL associativi Oracle in Amazon Aurora PostgreSQL e Amazon RDS per PostgreSQL
- Stima le dimensioni del motore Amazon RDS per un database Oracle utilizzando i report AWR
- Gestisci blocchi anonimi nelle istruzioni SQL dinamiche in Aurora PostgreSQL
- Migrazione incrementale da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL utilizzando Oracle SQL Developer e AWS SCT
- Carica i file BLOB in formato TEXT utilizzando la codifica dei file in Aurora, compatibile con PostgreSQL
- Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL in modalità SSL utilizzando AWS DMS
- Migrazione di un database Amazon RDS for Oracle su Account AWS un altro database Regione AWS e AWS DMS utilizzo per la replica continua
- Esegui la migrazione di un database Oracle locale su Amazon EC2 utilizzando Oracle Data Pump
- Esegui la migrazione di un database Oracle locale ad Amazon OpenSearch Service utilizzando Logstash
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for MySQL utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for Oracle
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for Oracle utilizzando l'importazione diretta di Oracle Data Pump tramite un collegamento al database
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for Oracle utilizzando Oracle Data Pump
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for PostgreSQL utilizzando un bystander Oracle e AWS DMS
- Esegui la migrazione di un database Oracle locale a Oracle su Amazon EC2

- Esegui la migrazione di un database Oracle da Amazon EC2 ad Amazon RDS for MariaDB utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database Oracle da Amazon EC2 ad Amazon RDS for Oracle utilizzando AWS DMS
- Esegui la migrazione di un database Oracle ad Amazon DynamoDB utilizzando AWS DMS
- Esegui la migrazione di un database Oracle ad Amazon RDS for Oracle utilizzando gli adattatori flat file GoldenGate Oracle
- Esegui la migrazione di un database Oracle ad Amazon Redshift utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database Oracle ad Aurora PostgreSQL utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database Oracle JD Edwards EnterpriseOne su AWS utilizzando Oracle Data Pump e AWS DMS
- Esegui la migrazione di una tabella partizionata Oracle su PostgreSQL utilizzando AWS DMS
- Esegui la migrazione di un PeopleSoft database Oracle su AWS utilizzando AWS DMS
- Esegui la migrazione dei dati da un database Oracle locale ad Aurora PostgreSQL
- Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for MySQL
- Esegui la migrazione da Oracle 8i o 9i ad Amazon RDS for PostgreSQL utilizzando viste materializzate e AWS DMS
- Esegui la migrazione da Oracle 8i o 9i ad Amazon RDS for PostgreSQL utilizzando AWS DMS SharePlex
- Esegui la migrazione da Oracle Database ad Amazon RDS for PostgreSQL utilizzando Oracle GoldenGate
- Esegui la migrazione da Oracle su Amazon EC2 ad Amazon RDS for MySQL utilizzando AWS DMS e AWS SCT
- Esegui la migrazione da Oracle WebLogic ad Apache Tomcat (ToMee) su Amazon ECS
- Migrazione di indici basati su funzioni da Oracle a PostgreSQL
- Migrazione delle applicazioni legacy da Oracle Pro*C a ECPG
- Esegui la migrazione dei valori Oracle CLOB su singole righe in PostgreSQL su AWS
- Esegui la migrazione dei codici di errore del database Oracle a un database compatibile con Amazon Aurora PostgreSQL
- Migrazione delle funzioni native di Oracle su PostgreSQL utilizzando le estensioni

- Esegui la migrazione PeopleSoft da Oracle ad Amazon RDS Custom
- Esegui la migrazione della funzionalità Oracle ROWID a PostgreSQL su AWS
- Migrazione dei pacchetti pragma Oracle SERIALLY_REUSABLE in PostgreSQL
- Migra le colonne virtuali generate da Oracle a PostgreSQL
- Monitora GoldenGate i log di Oracle utilizzando Amazon CloudWatch
- Analizza le dipendenze degli oggetti per le migrazioni parziali del database da Oracle a PostgreSQL
- Ripiattaforma Oracle Database Enterprise Edition alla Standard Edition 2 su Amazon RDS per Oracle
- Configurare un' HA/DR architettura per Oracle E-Business Suite su Amazon RDS Custom con un database di standby attivo
- Configura la funzionalità Oracle UTL_FILE su Aurora, compatibile con PostgreSQL
- Ruoli di transizione per un' PeopleSoft applicazione Oracle su Amazon RDS Custom for Oracle
- Convalida gli oggetti del database dopo la migrazione da Oracle ad Amazon Aurora PostgreSQL

SAP

- Esegui automaticamente il backup dei database SAP HANA utilizzando Systems Manager e EventBridge
- Esegui la migrazione da SAP ASE ad Amazon RDS per SQL Server utilizzando AWS DMS
- Esegui la migrazione di SAP ASE su Amazon Aurora, compatibile EC2 con PostgreSQL utilizzando AWS SCT e AWS DMS
- Esegui la migrazione da SAP HANA ad AWS utilizzando SAP HSR con lo stesso nome host
- Configura il disaster recovery per SAP su IBM Db2 su AWS

Altri modelli

- Accedi a un database Amazon Neptune da un container Amazon EKS
- Accedi, esegui query e unisciti a tabelle Amazon DynamoDB utilizzando Athena
- Consenti alle EC2 istanze l'accesso in scrittura ai bucket S3 negli account AMS
- Analizza e visualizza dati JSON annidati con Amazon Athena e Amazon Quick Sight
- Automatizza i backup per le istanze DB di Amazon RDS for PostgreSQL utilizzando AWS Batch
- Archivia automaticamente gli elementi su Amazon S3 utilizzando DynamoDB TTL
- Automatically remediate unencrypted Amazon RDS DB instances and clusters
- Arresta e avvia automaticamente un'istanza database Amazon RDS utilizzando AWS Systems Manager Maintenance Windows
- Crea una AWS landing zone che includa MongoDB Atlas
- Crea programmi COBOL Db2 usando e Modernizzazione del mainframe AWS AWS CodeBuild
- Crea una rete dati aziendale con Amazon DataZone e AWS CDK AWS CloudFormation
- Modifica le applicazioni Python e Perl per supportare la migrazione dei database da Microsoft SQL Server a Amazon Aurora PostgreSQL Compatible Edition
- Converti e decomprimi i dati EBCDIC in ASCII su AWS usando Python
- Convertire la funzionalità temporale Teradata NORMALIZE in Amazon Redshift SQL
- Convertire la funzionalità Teradata RESET WHEN in Amazon Redshift SQL
- Converti VARCHAR2 (1) il tipo di dati per Oracle in tipo di dati booleano per Amazon Aurora PostgreSQL
- Crea utenti e ruoli delle applicazioni in Aurora, compatibile con PostgreSQL
- Crea CloudFormation modelli AWS per attività AWS DMS utilizzando Microsoft Excel e Python
- Distribuisci i record DynamoDB ad Amazon S3 utilizzando Kinesis Data Streams e Firehose con AWS CDK
- Implementa un cluster Cassandra su Amazon EC2 con dati statici privati IPs per evitare il ribilanciamento
- Implementa un cluster CockroachDB in Amazon EKS utilizzando Terraform
- Sviluppa assistenti avanzati basati sull'intelligenza artificiale generativa utilizzando RAG e suggerimenti ReAct
- Emula Oracle DR utilizzando un database globale Aurora compatibile con PostgreSQL
- Abilita l'archiviazione dei DB2 log direttamente su Amazon S3 in un database IBM Db2

- Abilita la crittografia trasparente dei dati in Amazon RDS for SQL Server
- Esportazione di un database Microsoft SQL Server in Amazon S3 utilizzando AWS DMS
- SHA1 Implementa l'hashing per i dati PII durante la migrazione da SQL Server a PostgreSQL
- Migrazione incrementale da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL utilizzando Oracle SQL Developer e AWS SCT
- Carica i file BLOB in formato TEXT utilizzando la codifica dei file in Aurora, compatibile con PostgreSQL
- Gestisci le credenziali con AWS Secrets Manager
- Esegui la migrazione di un database Db2 da Amazon a EC2 Aurora compatibile con MySQL utilizzando AWS DMS
- Esegui la migrazione di un database Microsoft SQL Server da Amazon EC2 ad Amazon DocumentDB utilizzando AWS DMS
- Esegui la migrazione di un database Microsoft SQL Server su Aurora MySQL utilizzando AWS DMS e AWS SCT
- Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL in modalità SSL utilizzando AWS DMS
- Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL con e utilizzando e AWS SCT AWS DMS AWS CLI CloudFormation
- Esegui la migrazione di un'istanza database Amazon RDS su un altro VPC o account
- Migrazione di un database Amazon RDS for Oracle su Account AWS un altro database Regione AWS e AWS DMS utilizzo per la replica continua
- Esegui la migrazione di un cluster Amazon Redshift in una regione AWS in Cina
- Esegui la migrazione di un database MariaDB locale su Amazon RDS for MariaDB utilizzando strumenti nativi
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon EC2
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon RDS for SQL Server
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon RDS for SQL Server utilizzando server collegati
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon RDS for SQL Server utilizzando metodi di backup e ripristino nativi
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon Redshift utilizzando AWS DMS

- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon Redshift utilizzando gli agenti di estrazione dati AWS SCT
- Esegui la migrazione di un database Microsoft SQL Server locale a Microsoft SQL Server su Amazon EC2 con Linux
- Esegui la migrazione di un database MySQL locale su Amazon RDS for MySQL
- Esegui la migrazione di un database MySQL locale su Aurora MySQL
- Esegui la migrazione di un database Oracle locale su Amazon EC2 utilizzando Oracle Data Pump
- Esegui la migrazione di un database Oracle locale ad Amazon OpenSearch Service utilizzando Logstash
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for MySQL utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for Oracle
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for Oracle utilizzando l'importazione diretta di Oracle Data Pump tramite un collegamento al database
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for Oracle utilizzando Oracle Data Pump
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for PostgreSQL utilizzando un bystander Oracle e AWS DMS
- Esegui la migrazione di un database Oracle locale a Oracle su Amazon EC2
- Esegui la migrazione di un database PostgreSQL locale su Aurora PostgreSQL
- Esegui la migrazione di un database ThoughtSpot Falcon locale su Amazon Redshift
- Esegui la migrazione di un database Oracle da Amazon EC2 ad Amazon RDS for MariaDB utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database Oracle da Amazon EC2 ad Amazon RDS for Oracle utilizzando AWS DMS
- Esegui la migrazione di un database Oracle ad Amazon RDS for Oracle utilizzando gli adattatori flat file GoldenGate Oracle
- Esegui la migrazione di un database Oracle ad Amazon Redshift utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database Oracle ad Aurora PostgreSQL utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database Oracle JD Edwards EnterpriseOne su AWS utilizzando Oracle Data Pump e AWS DMS

- Esegui la migrazione di una tabella partizionata Oracle su PostgreSQL utilizzando AWS DMS
- Esegui la migrazione di un PeopleSoft database Oracle su AWS utilizzando AWS DMS
- Esegui la migrazione di un database Couchbase Server su Amazon EC2
- Esegui la migrazione dei dati da un database Oracle locale ad Aurora PostgreSQL
- Migrare i dati verso il Cloud AWS utilizzando Starburst
- Esegui la migrazione di Db2 for LUW ad Amazon EC2 utilizzando la spedizione di log per ridurre i tempi di interruzione
- Esegui la migrazione di Db2 per LUW ad Amazon EC2 con disaster recovery ad alta disponibilità
- Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for MySQL
- Migrazione da Couchbase Server a Couchbase Capella su AWS
- Esegui la migrazione da IBM Db2 su Amazon a EC2 Aurora PostgreSQL compatibile con AWS DMS e AWS SCT
- Esegui la migrazione da Oracle 8i o 9i ad Amazon RDS for PostgreSQL utilizzando viste materializzate e AWS DMS
- Esegui la migrazione da Oracle 8i o 9i ad Amazon RDS for PostgreSQL utilizzando AWS DMS SharePlex
- Esegui la migrazione da Oracle Database ad Amazon RDS for PostgreSQL utilizzando Oracle GoldenGate
- Esegui la migrazione da Oracle su Amazon EC2 ad Amazon RDS for MySQL utilizzando AWS DMS e AWS SCT
- Esegui la migrazione da PostgreSQL su Amazon RDS per PostgreSQL utilizzando EC2 pglogical
- Esegui la migrazione da SAP ASE ad Amazon RDS per SQL Server utilizzando AWS DMS
- Migrazione di indici basati su funzioni da Oracle a PostgreSQL
- Migrazione delle applicazioni legacy da Oracle Pro*C a ECPG
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon EC2 utilizzando Application Migration Service
- Esegui la migrazione dei carichi di lavoro Cloudera locali a Cloudera Data Platform su AWS
- Esegui la migrazione dei database MySQL locali su Aurora MySQL utilizzando Percona, Amazon EFS e Amazon S3 XtraBackup
- Esegui la migrazione di Oracle Business Intelligence 12c al cloud AWS dai server locali
- Esegui la migrazione dei valori Oracle CLOB su singole righe in PostgreSQL su AWS

- Esegui la migrazione dei codici di errore del database Oracle a un database compatibile con Amazon Aurora PostgreSQL
- Esegui la migrazione di tabelle esterne Oracle verso Amazon Aurora, compatibile con PostgreSQL
- Migrazione delle funzioni native di Oracle su PostgreSQL utilizzando le estensioni
- Esegui la migrazione PeopleSoft da Oracle ad Amazon RDS Custom
- Esegui la migrazione della funzionalità Oracle ROWID a PostgreSQL su AWS
- Migrazione dei pacchetti pragma Oracle SERIALLY_REUSABLE in PostgreSQL
- Esegui la migrazione di SAP ASE su Amazon Aurora, compatibile EC2 con PostgreSQL utilizzando AWS SCT e AWS DMS
- Migra le colonne virtuali generate da Oracle a PostgreSQL
- Imposta uno spazio dati minimo praticabile per condividere i dati tra le organizzazioni
- Monitora ElastiCache i cluster Amazon per la crittografia a riposo
- Interroga le tabelle Amazon DynamoDB con SQL utilizzando Amazon Athena
- Proteggi e semplifica l'accesso degli utenti in un database federativo Db2 su AWS utilizzando contesti affidabili
- Configura un' PeopleSoft architettura ad alta disponibilità su AWS
- Configura la funzionalità Oracle UTL_FILE su Aurora, compatibile con PostgreSQL
- Semplifica le implementazioni di PostgreSQL su Amazon EKS utilizzando PGO
- Trasferisci z/OS dati Db2 su larga scala su Amazon S3 in file CSV
- Trasporta i database PostgreSQL tra due istanze DB Amazon RDS utilizzando pg_transport
- Convalida gli oggetti del database dopo la migrazione da Oracle ad Amazon Aurora PostgreSQL

Archiviazione e backup

Argomenti

- Consenti alle EC2 istanze l'accesso in scrittura ai bucket S3 negli account AMS
- Automatizza l'inserimento di flussi di dati in un database Snowflake utilizzando Snowflake Snowpipe, Amazon S3, Amazon SNS e Amazon Data Firehose
- Crittografa automaticamente i volumi Amazon EBS esistenti e nuovi
- Eseguire il backup dei server Sun SPARC nell'emulatore Stromasys Charon-SSP sul Cloud AWS
- Esegui il backup e l'archiviazione dei dati su Amazon S3 con Veeam Backup & Replication
- Copia i dati da un bucket Amazon S3 a un altro account e regione utilizzando AWS CLI
- Abilita l'archiviazione dei DB2 log direttamente su Amazon S3 in un database IBM Db2
- Esegui la migrazione dei dati da un ambiente Hadoop locale ad Amazon S3 utilizzando AWS per Amazon S3 DistCp PrivateLink
- Altri modelli

Consenti alle EC2 istanze l'accesso in scrittura ai bucket S3 negli account AMS

Mansi Suratwala, Amazon Web Services

Riepilogo

AWS Managed Services (AMS) ti aiuta a gestire la tua AWS infrastruttura in modo più efficiente e sicuro. Gli account AMS dispongono di protezioni di sicurezza per l'amministrazione standardizzata delle risorse. AWS Un ostacolo è che i profili di istanza predefiniti di Amazon Elastic Compute Cloud EC2 (Amazon) non consentono l'accesso in scrittura ai bucket Amazon Simple Storage Service (Amazon S3). Tuttavia, la tua organizzazione potrebbe avere più bucket S3 e richiedere un maggiore controllo sull'accesso per istanze. EC2 Ad esempio, potresti voler archiviare i backup del database dalle EC2 istanze in un bucket S3.

Questo schema spiega come utilizzare le request for change (RFCs) per consentire alle EC2 istanze l'accesso in scrittura ai bucket S3 nel tuo account AMS. Una RFC è una richiesta creata da te o da AMS per apportare una modifica al tuo ambiente gestito e che include un ID del tipo di modifica (CT) per una particolare operazione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AMS Advanced. Per ulteriori informazioni a riguardo, consulta i piani operativi AMS nella documentazione AMS.
- Accesso al `customer-mc-user-role` ruolo AWS Identity and Access Management (IAM) da inviare RFCs.
- AWS Command Line Interface (AWS CLI), installato e configurato con le EC2 istanze del tuo account AMS.
- Comprensione di come creare e inviare RFCs in AMS. Per ulteriori informazioni su questo argomento, consulta Cosa sono i tipi di modifica AMS? nella documentazione AMS.
- Comprensione dei tipi di modifica manuali e automatizzati (CTs). Per ulteriori informazioni su questo argomento, consulta Automatizzato e manuale CTs nella documentazione AMS.

Architecture

Stack tecnologico

- ARMS
- AWS CLI
- Amazon EC2
- Simple Storage Service (Amazon S3)
- IAM

Tools (Strumenti)

- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire Servizi AWS tramite comandi nella shell della riga di comando.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Managed Services (AMS) ti aiuta a gestire la tua infrastruttura AWS in modo più efficiente e sicuro.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.

Epiche

Crea un bucket S3 con un RFC

Operazione	Description	Competenze richieste
Crea un bucket S3 utilizzando un RFC automatizzato.	1. Accedi al tuo account AMS, scegli la pagina Scegli il tipo di modifica, scegli RFCs, quindi scegli Crea RFC. 2. Invia la RFC automatizzata Create S3 Bucket.	Amministratore di sistema AWS, sviluppatore AWS

Operazione	Description	Competenze richieste
	Note Assicurati di registrare il nome del bucket S3.	

Crea un profilo di istanza IAM e associalo alle EC2 istanze

Operazione	Description	Competenze richieste
Invia una RFC manuale per creare un ruolo IAM.	Quando viene effettuato l'onboarding di un account AMS, `customer-mc-ec2-instance-profile` viene creato un profilo di istanza IAM predefinito denominato e associato a ciascuna EC2 istanza del tuo account AMS. Tuttavia, il profilo dell'istanza non dispone delle autorizzazioni di scrittura per i bucket S3. Per aggiungere i permessi di scrittura, invia il manuale RFC Create IAM Resource per creare un ruolo IAM con le seguenti tre politiche:, e. `customer_ec2_instance_` `customer_deny_policy` `customer_ec2_s3_integration_policy`	Amministratore di sistema AWS, sviluppatore AWS

Operazione	Description	Competenze richieste
	Important Le `customer_deny_policy` politiche `customer_ec2_instance_` and esistono già nel tuo account AMS. Tuttavia, è necessario creare `customer_ec2_s3_integration_policy` utilizzando la seguente politica di esempio:	

```
{
  "Version": "2012-10-17",
   "Statement": [
    {
      "Sid": "",
       "Effect": "Allow",
       "Principal": {
         "Service": "ec2.amazonaws.com"
      },
       "Action": "sts:AssumeRole"
    }
  ]
}

Role Permissions:
{
```

Operazione	Description	Competenze richieste
	"Version": "2012-10-17", "Statement": [{ "Action": ["s3:ListBucket", "s3:GetBucketLocation"], "Resource": "arn:aws:s3:::", "Effect": "Allow" }, { "Action": ["s3:GetObject", "s3:PutObject", "s3:ListMultipartUploadParts", "s3:AbortMultipartUpload"], "Resource": "arn:aws:s3:::/*", "Effect": "Allow" }] }	

Operazione	Description	Competenze richieste
Invia una RFC manuale per sostituire il profilo dell'istanza IAM.	Invia una RFC manuale per associare le EC2 istanze di destinazione al nuovo profilo di istanza IAM.	Amministratore di sistema AWS, sviluppatore AWS
Prova un'operazione di copia nel bucket S3.	Prova un'operazione di copia nel bucket S3 eseguendo il seguente comando in: AWS CLI `aws s3 cp test.txt s3://<S3 bucket>/test2.txt`	Amministratore di sistema AWS, sviluppatore AWS

Risorse correlate

- Crea un profilo di istanza IAM per le tue EC2 istanze Amazon
- Creazione di un bucket S3 (utilizzando la console AWS SDKs Amazon S3 o) AWS CLI

Automatizza l'inserimento di flussi di dati in un database Snowflake utilizzando Snowflake Snowpipe, Amazon S3, Amazon SNS e Amazon Data Firehose

Bikash Chandra Rout, Amazon Web Services

Riepilogo

Questo modello descrive come utilizzare i servizi su Amazon Web Services (AWS) Cloud per elaborare un flusso continuo di dati e caricarlo in un database Snowflake. Il modello utilizza Amazon Data Firehose per inviare i dati ad Amazon Simple Storage Service (Amazon S3), Amazon Simple Notification Service (Amazon SNS) per inviare notifiche quando vengono ricevuti nuovi dati e Snowflake Snowpipe per caricare i dati in un database Snowflake.

Seguendo questo schema, puoi avere a disposizione i dati generati continuamente per l'analisi in pochi secondi, evitare più `COPY` comandi manuali e avere il supporto completo per i dati semistrutturati in fase di caricamento.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS.
- Una fonte di dati che invia continuamente dati a un flusso di distribuzione Firehose.
- Un bucket S3 esistente che riceve i dati dal flusso di distribuzione di Firehose.
- Un account Snowflake attivo.

Limitazioni

- Snowflake Snowpipe non si collega direttamente a Firehose.

Architecture

Stack tecnologico

- Amazon Data Firehose

- Amazon SNS
- Simple Storage Service (Amazon S3)
- Snowflake Snowpipe
- Banca dati Snowflake

Tools (Strumenti)

- Amazon Data Firehose è un servizio completamente gestito per la distribuzione di dati di streaming in tempo reale a destinazioni come Amazon S3, Amazon Redshift, OpenSearch Amazon Service, Splunk e qualsiasi endpoint HTTP o endpoint HTTP personalizzato di proprietà di provider di servizi terzi supportati.
- Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è uno storage per Internet.
- Amazon Simple Notification Service (Amazon SNS) coordina e gestisce la consegna o l'invio di messaggi agli endpoint o ai clienti abbonati.
- Snowflake — Snowflake è un data warehouse analitico fornito come (SaaS). Software-as-a-Service
- Snowflake Snowpipe: Snowpipe carica i dati dai file non appena sono disponibili in una fase Snowflake.

Epiche

Configura uno Snowflake Snowpipe

Operazione	Description	Competenze richieste
Crea un file CSV in Snowflake.	Accedi a Snowflake ed esegui il `CREATE FILE FORMAT` comando per creare un file CSV con un delimitatore di campo specificato. Per ulteriori informazioni su questo e altri comandi Snowflake, consulta la sezione Informazioni aggiuntive.	Developer

Operazione	Description	Competenze richieste
Crea uno stage Snowflake esterno.	Eseguite il `CREATE STAGE` comando per creare uno stage Snowflake esterno che faccia riferimento al file CSV creato in precedenza. Importante: avrai bisogno dell'URL del bucket S3, della tua chiave di accesso e della tua chiave di AWS accesso segreta. AWS Esegui il `SHOW STAGES` comando per verificare che lo stage Snowflake sia stato creato.	Developer
Create la tabella di destinazi one Snowflake.	Esegui il `CREATE TABLE` comando per creare la tabella Snowflake.	Developer
Crea una pipa.	Esegui il `CREATE PIPE` comando; assicurati che `auto_ingest=true` sia nel comando. Eseguite il `SHOW PIPES` comando per verificare che la pipe sia stata creata. Copia e salva il valore `notification_channel` della colonna. Questo valore verrà utilizzato per configura re le notifiche degli eventi di Amazon S3.	Developer

Configura il bucket S3

Operazione	Description	Competenze richieste
Crea una politica del ciclo di vita di 30 giorni per il bucket S3.	Accedi Console di gestione AWS e apri la console Amazon S3. Scegliete il bucket S3 che contiene i dati di Firehose. Quindi scegli la scheda Gestione nel bucket S3 e scegli Aggiungi regola del ciclo di vita. Inserisci un nome per la regola nella finestra di dialogo delle regole del ciclo di vita e configura una regola del ciclo di vita di 30 giorni per il tuo bucket. Per informazioni su questa e altre storie, consulta la sezione Risorse correlate.	Amministratore di sistema, sviluppatore
Crea una policy IAM per il bucket S3.	Apri la console AWS Identity and Access Managemen t (IAM) e scegli Policies. Scegli Create policy (Crea policy), quindi scegli la scheda JSON. Copia e incolla la policy dalla sezione Informazioni aggiuntive nel campo JSON. Questa politica concederà `PutObject` e `DeleteObj ect` autorizzazioni, nonché `GetObject` `GetObject Version` , e `ListBucket` autorizzazioni. Scegli Rivedi politica, inserisci un nome di	Amministratore di sistema, sviluppatore

Operazione	Description	Competenze richieste
	politica, quindi scegli Crea politica.	
Assegna la policy a un ruolo IAM.	Apri la console IAM, scegli Ruoli, quindi scegli Crea ruolo. Scegli un altro account AWS come entità affidabile. Inserisci il tuo Account AWS ID e scegli Richiedi un ID esterno. Inserisci un ID segnaposto che potrai modificare in seguito. Scegli Avanti e assegna la policy IAM che hai creato in precedenza. Quindi crea il ruolo IAM.	Amministratore di sistema, sviluppatore
Copia l'Amazon Resource Name (ARN) per il ruolo IAM.	Apri la console IAM e scegli Ruoli. Scegli il ruolo IAM che hai creato in precedenza, quindi copia e archivia l'ARN del ruolo.	Amministratore di sistema, sviluppatore

Configura un'integrazione dello storage in Snowflake

Operazione	Description	Competenze richieste
Crea un'integrazione di archiviazione in Snowflake.	Accedi a Snowflake ed esegui il comando. `CREATE STORAGE INTEGRATION` Ciò modificherà la relazione di fiducia, concederà l'accesso a Snowflake e fornirà l'ID esterno per il tuo stage Snowflake.	Amministratore di sistema, sviluppatore

Operazione	Description	Competenze richieste
Recupera il ruolo IAM per il tuo account Snowflake.	Esegui il `DESC INTEGRATION` comando per recuperare l'ARN per il ruolo IAM. ⚠ Important `<integration_name>` è il nome dell'integrazione di storage Snowflake creata in precedenza.	Amministratore di sistema, sviluppatore
Registra due valori di colonna.	Copia e salva i valori per le `storage_aws_external_id` colonne `storage_aws_iam_user_arn` and.	Amministratore di sistema, sviluppatore

Consenti a Snowflake Snowpipe di accedere al bucket S3

Operazione	Description	Competenze richieste
Modifica la politica dei ruoli IAM.	Apri la console IAM e scegli Ruoli. Scegli il ruolo IAM che hai creato in precedenza e scegli la scheda Relazioni di fiducia. Seleziona Modifica relazione di attendibilità. `snowflake_external_id` Sostituiscilo con il `storage_aws_external_id` valore che hai copiato in precedenza. Sostituisci `snowflake_user_arn` con il `storage_a`	Amministratore di sistema, sviluppatore

Operazione	Description	Competenze richieste
	`ws_iam_user_arn` valore che hai copiato in precedenza. Quindi scegli Aggiorna politica di fiducia.	

Attiva e configura le notifiche SNS per il bucket S3

Operazione	Description	Competenze richieste
Attiva le notifiche degli eventi per il bucket S3.	Apri la console Amazon S3 e scegli il tuo bucket. Scegli Proprietà e, in Impostazioni avanzate, scegli Eventi. Scegli Aggiungi notifica e inserisci un nome per questo evento. Se non inserisci un nome, verrà utilizzato un identificatore univoco globale (GUID).	Amministratore di sistema, sviluppatore
Configura le notifiche Amazon SNS per il bucket S3.	In Eventi, scegli ObjectCre ate (Tutti), quindi scegli SQS Queue nell'elenco a discesa Invia a. Nell'elenco SNS, scegli Aggiungi ARN della coda SQS e incolla `notification_chann el` il valore che hai copiato in precedenza. Quindi scegli Save (Salva).	Amministratore di sistema, sviluppatore
Sottoscrivi la coda Snowflake SQS all'argomento SNS.	Sottoscrivi la coda Snowflake SQS all'argomento SNS che hai creato. Per informazioni su	Amministratore di sistema, sviluppatore

Operazione	Description	Competenze richieste
	questo passaggio, consulta la sezione Risorse correlate.	

Verifica l'integrazione con Snowflake Stage

Operazione	Description	Competenze richieste
Controlla e prova Snowpipe.	Accedi a Snowflake e apri il livello Snowflake. Trascina i file nel tuo bucket S3 e controlla se la tabella Snowflake li carica. Amazon S3 invierà notifiche SNS a Snowpipe quando vengono visualizzati nuovi oggetti nel bucket S3.	Amministratore di sistema, sviluppatore

Risorse correlate

- Gestione del ciclo di vita dello storage
- Sottoscrivi la coda SQS di Snowflake all'argomento Amazon SNS

Informazioni aggiuntive

Crea un formato di file:

```
CREATE FILE FORMAT <name>
TYPE = 'CSV'
FIELD_DELIMITER = '|'
SKIP_HEADER = 1;
```

Crea una fase esterna:

```
externalStageParams (for Amazon S3) ::=
```

```
  URL = 's3://[//]'

  [ { STORAGE_INTEGRATION =  } | { CREDENTIALS = ( {  { AWS_KEY_ID = `` AWS_SECRET_KEY
 = `` [ AWS_TOKEN = `` ] } | AWS_ROLE = ``  } ) ) }` ]
  [ ENCRYPTION = ( [ TYPE = 'AWS_CSE' ] [ MASTER_KEY = '' ] |
                   [ TYPE = 'AWS_SSE_S3' ] |
                   [ TYPE = 'AWS_SSE_KMS' [ KMS_KEY_ID = '' ] |
                   [ TYPE = NONE ] )
```

Crea una tabella:

```
CREATE [ OR REPLACE ] [ { [ LOCAL | GLOBAL ] TEMP[ORARY] | VOLATILE } | TRANSIENT ]
 TABLE [ IF NOT EXISTS ]
  <table_name>
    ( <col_name> <col_type> [ { DEFAULT <expr>
                                | { AUTOINCREMENT | IDENTITY } [ ( <start_num> ,
 <step_num> ) | START <num> INCREMENT <num> ] } ]
                                 /* AUTOINCREMENT / IDENTITY supported only for numeric
 data types (NUMBER, INT, etc.) */
                              [ inlineConstraint ]
      [ , <col_name> <col_type> ... ]
      [ , outoflineConstraint ]
      [ , ... ] )
  [ CLUSTER BY ( <expr> [ , <expr> , ... ] ) ]
  [ STAGE_FILE_FORMAT = ( { FORMAT_NAME = '<file_format_name>'
                           | TYPE = { CSV | JSON | AVRO | ORC | PARQUET | XML }
 [ formatTypeOptions ] } ) ]
  [ STAGE_COPY_OPTIONS = ( copyOptions ) ]
  [ DATA_RETENTION_TIME_IN_DAYS = <num> ]
  [ COPY GRANTS ]
  [ COMMENT = '<string_literal>' ]
```

Mostra fasi:

```
SHOW STAGES;
```

Crea una pipa:

```
CREATE [ OR REPLACE ] PIPE [ IF NOT EXISTS ]
  [ AUTO_INGEST = [ TRUE | FALSE ] ]
  [ AWS_SNS_TOPIC =  ]
  [ INTEGRATION = '' ]
```

```
  [ COMMENT = '' ]
  AS
```

Mostra tubi:

```
SHOW PIPES [ LIKE '<pattern>' ]
           [ IN { ACCOUNT | [ DATABASE ] <db_name> | [ SCHEMA ] <schema_name> } ]
```

Crea un'integrazione di archiviazione:

```
CREATE STORAGE INTEGRATION <integration_name>
  TYPE = EXTERNAL_STAGE
  STORAGE_PROVIDER = S3
  ENABLED = TRUE
  STORAGE_AWS_ROLE_ARN = '<iam_role>'
  STORAGE_ALLOWED_LOCATIONS = ('s3://<bucket>/<path>/', 's3://<bucket>/<path>/')
  [ STORAGE_BLOCKED_LOCATIONS = ('s3://<bucket>/<path>/', 's3://<bucket>/<path>/') ]
```

Esempio:

```
create storage integration s3_int
  type = external_stage
  storage_provider = s3
  enabled = true
  storage_aws_role_arn = 'arn:aws:iam::001234567890:role/myrole'
  storage_allowed_locations = ('s3://amzn-s3-demo-bucket1/mypath1/', 's3://amzn-s3-
demo-bucket2/mypath2/')
  storage_blocked_locations = ('s3://amzn-s3-demo-bucket1/mypath1/sensitivedata/',
 's3://amzn-s3-demo-bucket2/mypath2/sensitivedata/');
```

Per ulteriori informazioni su questo passaggio, consulta Configurazione di un'integrazione di storage Snowflake per accedere ad Amazon S3 dalla documentazione di Snowflake.

Descrivi un'integrazione:

```
DESC INTEGRATION <integration_name>;
```

Politica sui bucket S3:

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
              "s3:PutObject",
              "s3:GetObject",
              "s3:GetObjectVersion",
              "s3:DeleteObject",
              "s3:DeleteObjectVersion"
            ],
            "Resource": "arn:aws:s3:::/*"
        },
        {
            "Effect": "Allow",
            "Action": "s3:ListBucket",
            "Resource": "arn:aws:s3:::",
            "Condition": {
                "StringLike": {
                    "s3:prefix": [
                        "/*"
                    ]
                }
            }
        }
    ]
}
```

Crittografa automaticamente i volumi Amazon EBS esistenti e nuovi

Tony DeMarco e Josh Joy, Amazon Web Services

Riepilogo

La crittografia dei volumi di Amazon Elastic Block Store (Amazon EBS) è importante per la strategia di protezione dei dati di un'organizzazione. È un passo importante nella creazione di un ambiente ben architettato. Sebbene non esista un modo diretto per crittografare i volumi o le istantanee EBS non crittografati esistenti, è possibile crittografarli creando un nuovo volume o un'istantanea. Per ulteriori informazioni, consulta le risorse di Encrypt EBS nella documentazione di Amazon EC2 . Questo modello fornisce controlli preventivi e investigativi per crittografare i volumi EBS, sia nuovi che esistenti. In questo modello, si configurano le impostazioni dell'account, si creano processi di riparazione automatizzati e si implementano i controlli di accesso.

Prerequisiti e limitazioni

Prerequisiti

- Un account Amazon Web Services (AWS) attivo
- AWS Command Line Interface (AWS CLI), installata e configurata su macOS, Linux o Windows
- jq, installato e configurato su macOS, Linux o Windows
- Vengono fornite le autorizzazioni AWS Identity and Access Management (IAM) per avere accesso in lettura e scrittura ad AWS, CloudFormation Amazon Elastic Compute Cloud (Amazon EC2), AWS Systems Manager, AWS Config e AWS Key Management Service (AWS KMS)
- AWS Organizations è configurato con tutte le funzionalità abilitate, un requisito per le politiche di controllo del servizio
- AWS Config è abilitato negli account di destinazione

Limitazioni

- Nell'account AWS di destinazione non devono esserci regole AWS Config denominate encrypted-volumes. Questa soluzione implementa una regola con questo nome. Le regole preesistenti con questo nome possono causare il fallimento della distribuzione e comportare costi inutili relativi all'elaborazione della stessa regola più di una volta.
- Questa soluzione crittografa tutti i volumi EBS con la stessa chiave AWS KMS.

- Se abiliti la crittografia dei volumi EBS per l'account, questa impostazione è specifica della regione. Se lo abiliti per una regione AWS, non puoi disabilitarlo per singoli volumi o snapshot in quella regione. Per ulteriori informazioni, consulta Encryption by default nella EC2 documentazione di Amazon.
- Quando ripristini volumi EBS esistenti non crittografati, assicurati che l' EC2 istanza non sia in uso. Questa automazione spegne l'istanza per scollegare il volume non crittografato e collegare quello crittografato. Si verificano tempi di inattività durante la riparazione. Se si tratta di un'infrastruttura fondamentale per la tua organizzazione, assicurati che siano presenti configurazioni manuali o automatiche ad alta disponibilità in modo da non influire sulla disponibilità delle applicazioni in esecuzione sull'istanza. Si consiglia di ripristinare le risorse critiche solo durante le finestre di manutenzione standard.

Architecture

Workflow di automazione

1. AWS Config rileva un volume EBS non crittografato.
2. Un amministratore utilizza AWS Config per inviare un comando di riparazione a Systems Manager.
3. L'automazione Systems Manager scatta un'istantanea del volume EBS non crittografato.
4. L'automazione Systems Manager utilizza AWS KMS per creare una copia crittografata dello snapshot.
5. L'automazione Systems Manager esegue le seguenti operazioni:
   a. Arresta l' EC2 istanza interessata se è in esecuzione
   b. Allega la nuova copia crittografata del volume all'istanza EC2
   c. Riporta l' EC2 istanza allo stato originale

Tools (Strumenti)

Servizi AWS

- AWS CLI: l'AWS Command Line Interface (AWS CLI) fornisce accesso diretto alle interfacce di programmazione delle applicazioni pubbliche APIs () dei servizi AWS. Puoi esplorare le funzionalità di un servizio con l'AWS CLI e sviluppare script di shell per gestire le tue risorse. Oltre ai comandi equivalenti alle API di basso livello, diversi servizi AWS forniscono personalizzazioni per l'AWS

CLI. Le personalizzazioni possono includere comandi di livello più elevato che semplificano l'utilizzo di un servizio con un'API complessa.

- AWS CloudFormation: AWS CloudFormation è un servizio che ti aiuta a modellare e configurare le tue risorse AWS. Crei un modello che descrive tutte le risorse AWS che desideri (come EC2 le istanze Amazon) e fornisce e CloudFormation configura tali risorse per te.
- AWS Config: AWS Config fornisce una visualizzazione dettagliata della configurazione delle risorse AWS nel tuo account AWS. Questo include le relazioni tra le risorse e la maniera in cui sono state configurate in passato, in modo che tu possa vedere come le configurazioni e le relazioni cambiano nel corso del tempo.
- Amazon EC2 — Amazon Elastic Compute Cloud (Amazon EC2) è un servizio Web che fornisce una capacità di calcolo ridimensionabile da utilizzare per creare e ospitare i sistemi software.
- AWS KMS — AWS Key Management Service (AWS KMS) è un servizio di crittografia e gestione delle chiavi scalato per il cloud. Le chiavi e le funzionalità di AWS KMS vengono utilizzate da altri servizi AWS e puoi utilizzarle per proteggere i dati nel tuo ambiente AWS.
- AWS Organizations — AWS Organizations è un servizio di gestione degli account che consente di consolidare più account AWS in un'organizzazione da creare e gestire centralmente.
- AWS Systems Manager Automation — Systems Manager Automation semplifica le attività comuni di manutenzione e distribuzione per EC2 le istanze Amazon e altre risorse AWS.

Altri servizi

- jq — jq è un processore JSON a riga di comando leggero e flessibile. Questo strumento viene utilizzato per estrarre informazioni chiave dall'output della CLI di AWS.

Codice

- Il codice per questo pattern è disponibile nell'archivio delle chiavi KMS dei clienti per la correzione GitHub automatica dei volumi EBS non crittografati.

Epiche

Automatizza la riparazione dei volumi non crittografati

Operazione	Description	Competenze richieste
Scarica script e CloudFormation modelli.	Scarica lo script di shell, il file JSON e i CloudFormation modelli dall'archivio di chiavi KMS del GitHub cliente per la correzione automatica dei volumi EBS non crittografati.	Amministratore AWS, AWS generale
Identifica l'amministratore per la chiave AWS KMS.	1. Accedi alla Console di gestione AWS, quindi apri la console di IAM all'indirizzo https://console.aws.amazon.com/iam/. 2. Identifica un utente o un ruolo che sarà l'amministratore chiave di AWS KMS. Se è necessario creare un nuovo utente o ruolo per questo scopo, crealo subito. Per ulteriori informazioni, consulta IAM Identities nella documentazione IAM. Questa automazione crea una nuova chiave AWS KMS. 3. Una volta identificato, copia l'Amazon Resource Name (ARN) dell'utente o del ruolo. Per ulteriori informazioni, consulta IAM ARNs nella documentazione IAM.	Amministratore AWS, AWS generale

Operazione	Description	Competenze richieste
	Utilizzerai questo ARN nel passaggio successivo.	

Operazione	Description	Competenze richieste
Implementa il modello Stack1 CloudFormation .	1. Apri la CloudFormation console AWS all'indirizzo https://console.aws.amazon.com/cloudformation/. 2. In CloudFormation, distribuisci il `Stack1.yaml` modello. Tieni presente i seguenti dettagli di distribuzione: • Assegna allo stack un nome chiaro e descrittivo. Annota il nome dello stack perché avrai bisogno di questo valore nel passaggio successivo. • Incolla l'ARN dell'amministratore delle chiavi nell'unico campo del parametro in Stack1. Questo utente o ruolo diventa l'amministratore della chiave AWS KMS creata dallo stack. Per ulteriori informazioni sulla distribuzione di un CloudFormation modello, consulta Working with AWS CloudFormation templates nella CloudFormation documentazione.	Amministratore AWS, AWS generale

Operazione	Description	Competenze richieste
Implementa il modello Stack2 CloudFormation .	In CloudFormation, distribuisci il modello. `Stack2.yaml` Tieni presente i seguenti dettagli di distribuzione: • Assegna allo stack un nome chiaro e descrittivo. • Per l'unico parametro di Stack2, inserisci il nome dello stack che hai creato nel passaggio precedente. Ciò consente a Stack2 di fare riferimento alla nuova chiave e al ruolo AWS KMS distribuiti dallo stack nella fase precedente.	Amministratore AWS, AWS generale
Crea un volume non crittografato per il test.	Crea un' EC2 istanza con un volume EBS non crittografato. Per istruzioni, consulta Creare un volume Amazon EBS nella EC2 documentazione di Amazon. Il tipo di istanza non ha importanza e l'accesso all'istanza non è necessario. Puoi creare un'istanza t2.micro per rimanere nel livello gratuito e non è necessario creare una key pair.	Amministratore AWS, AWS generale

Operazione	Description	Competenze richieste
Testa la regola AWS Config.	1. Apri la console AWS Config all'indirizzo. https://console.aws.amazon.com/config/ Nella pagina Regole, scegli la regola dei volumi crittografati. 2. Verifica che la tua nuova istanza di test non crittografata compaia nell'elenco delle risorse non conformi. Se il volume non viene visualizzato immediatamente, attendi qualche minuto e aggiorna i risultati. La regola AWS Config rileva le modifiche alle risorse subito dopo la creazione dell'istanza e del volume. 3. Seleziona la risorsa, quindi scegli Remediate. È possibile visualizzare l'avanzamento e lo stato della riparazione in Systems Manager come segue: 1. Apri la console AWS Systems Manager all'indirizzo https://console.aws.amazon.com/systems-manager/. 2. Nel pannello di navigazione, scegli Automazione.	Amministratore AWS, AWS generale

Operazione	Description	Competenze richieste
	3. Scegli il link Execution ID per visualizzare i passaggi e lo stato.	
Configura account o regioni AWS aggiuntivi.	Se necessario per il tuo caso d'uso, ripeti questa epopea per eventuali account o regioni AWS aggiuntivi.	Amministratore AWS, AWS generale

Abilita la crittografia a livello di account dei volumi EBS

Operazione	Description	Competenze richieste
Esegui lo script di abilitazione.	1. In una shell bash, usa il `cd` comando per navigare nel repository clonato. 2. Inserisci il comando seguente per eseguire lo script `enable-ebs-encryption-for-account` . `./Bash/enable-ebs-encryption-for-account.sh`	Amministratore AWS, AWS generale, bash
Conferma che le impostazioni siano aggiornate.	1. Apri la EC2 console Amazon all'indirizzo https://console.aws.amazon.com/ec2/. 2. Sul lato destro dello schermo, in Impostazioni, scegli Protezione e sicurezza dei dati.	Amministratore AWS, AWS generale

Operazione	Description	Competenze richieste
	3. Nella sezione Crittografia EBS, conferma che l'opzione Crittografa sempre nuovi volumi EBS sia attivata e che la chiave di crittografia predefinita sia impostata sull'ARN specificato in precedenza. **Note** Se l'impostazione Always encrypt new EBS volumes è disattivata o la chiave è ancora impostata su alias/aws/ebs, conferma di aver effettuato l'accesso allo stesso account e alla stessa regione AWS in cui hai eseguito lo script di shell e controlla la presenza di messaggi di errore nella shell.	
Configura account o regioni AWS aggiuntivi.	Se necessario per il tuo caso d'uso, ripeti questa epopea per eventuali account o regioni AWS aggiuntivi.	Amministratore AWS, AWS generale

Impedisci la creazione di istanze non crittografate

Operazione	Description	Competenze richieste
Crea una politica di controllo del servizio.	1. Apri la console AWS Organizations nella versione https://console.aws.amazon.com/organizations/v2/. 2. Crea una nuova policy di controllo dei servizi. Per ulteriori informazioni, consulta Creazione di una policy di controllo dei servizi nella documentazione di AWS Organizations. 3. Aggiungi il contenuto di `DenyUnencryptedEC2.json` alla policy e salvalo. Hai scaricato questo file JSON dal GitHub repository nella prima epic. 4. Allega questa politica alla radice dell'organizzazione o a qualsiasi unità organizzativa necessaria (). OUs Per ulteriori informazioni, consulta Allegare e scollegare le policy di controllo del servizio nella documentazione di AWS Organizations.	Amministratore AWS, AWS generale

Risorse correlate

Documentazione del servizio AWS

- AWS CLI
- AWS Config
- AWS CloudFormation
- Amazon EC2
- AWS KMS
- AWS Organizations
- AWS Systems Manager Automation

Altre risorse

- manuale jq (sito web jq)
- scarica jq () GitHub

Eseguire il backup dei server Sun SPARC nell'emulatore Stromasys Charon-SSP sul Cloud AWS

Kevin Yung e Rohit Darji, Amazon Web Services

Luis Ramos, Stromasys

Riepilogo

Questo modello offre quattro opzioni per il backup dei server SPARC di Sun Microsystems dopo una migrazione da un ambiente locale al cloud Amazon Web Services ().AWS Queste opzioni di backup consentono di implementare un piano di backup che soddisfi il Recovery Point Objective (RPO) e il Recovery Time Objective (RTO) dell'organizzazione, utilizzi approcci automatizzati e riduca i costi operativi complessivi. Il modello fornisce una panoramica delle quattro opzioni di backup e dei passaggi per implementarle.

Se si utilizza un server Sun SPARC ospitato come guest su un emulatore Stromasys Charon-SSP, è possibile utilizzare una delle tre opzioni di backup seguenti:

- Opzione di backup 1: nastro virtuale Stromasys: utilizza la funzionalità di nastro virtuale Charon-SSP per configurare una struttura di backup nel server Sun SPARC e archiviare i file di backup su Amazon Simple Storage Service (Amazon S3) utilizzando Automation.AWS Systems Manager
- Opzione di backup 2: istantanea Stromasys: utilizza la funzionalità snapshot Charon-SSP per configurare una struttura di backup per i server guest Sun SPARC in Charon-SSP.
- Opzione di backup 3: istantanea del volume Amazon Elastic Block Store (Amazon EBS) — Se ospiti l'emulatore Charon-SSP su Amazon Elastic Compute Cloud (Amazon EC2), puoi utilizzare uno snapshot del volume Amazon EBS per creare backup per un file system Sun SPARC.

Se utilizzi un server Sun SPARC ospitato come guest su hardware e Charon-SSP su Amazon EC2, puoi utilizzare la seguente opzione di backup:

- Opzione di backup 4: libreria a nastro Gateway di archiviazione AWS virtuale (VTL): utilizza un'applicazione di backup con un gateway a nastro VTL Storage Gateway per eseguire il backup dei server Sun SPARC.

Se si utilizza un server Sun SPARC ospitato come zona brandizzata in un server Sun SPARC, è possibile utilizzare le opzioni di backup 1, 2 e 4.

Stromasys fornisce software e servizi per emulare i sistemi critici SPARC, Alpha, VAX e PA-RISC esistenti. Per ulteriori informazioni sulla migrazione all' Cloud AWS utilizzo dell'emulazione Stromasys, vedi Rehosting di SPARC, Alpha o altri sistemi legacy to with Stromasys sul blog. AWS AWS

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Server Sun SPARC esistenti.
- Licenze esistenti per Charon-SSP. Le licenze per Charon-SSP sono disponibili presso Stromasys Virtual Environment (VE) Marketplace AWS e le licenze per Stromasys Virtual Environment (VE) sono disponibili presso Stromasys. Per ulteriori informazioni, contattate il reparto vendite di Stromasys.
- Familiarità con i server Sun SPARC e i backup Linux.
- Familiarità con la tecnologia di emulazione Charon-SSP. Per ulteriori informazioni su questo argomento, consulta l'emulazione del server legacy di Stromasys nella documentazione di Stromasys.
- Se si desidera utilizzare la struttura a nastro virtuale o le applicazioni di backup per i file system dei server Sun SPARC, è necessario creare e configurare le strutture di backup per il file system del server Sun SPARC.
- Comprensione di RPO e RTO. Per ulteriori informazioni su questo argomento, consulta gli obiettivi di disaster recovery tratti dal white paper Reliability Pillar nella documentazione di Well-Architected AWS Framework.
- Per utilizzare l'opzione di Backup 4, è necessario disporre di quanto segue:
  - Un'applicazione di backup basata su software che supporta uno Storage Gateway VTL Tape Gateway. Per ulteriori informazioni su questo argomento, vedere Lavorare con i dispositivi VTL nella documentazione. Gateway di archiviazione AWS
  - Bacula Director o un'applicazione di backup simile, installata e configurata. Per ulteriori informazioni su questo argomento, consulta la documentazione di Bacula Director.

La tabella seguente fornisce informazioni sulle quattro opzioni di backup presenti in questo modello.

Opzioni di backup	Raggiunge la coerenza in caso di crash?	Raggiunge la coerenza delle applicazioni?	Soluzione di appliance di backup virtuale?	Casi d'uso tipici
Opzione 1: nastro virtuale Stromasys	Sì È possibile automatizzare le istantanee del file system Sun SPARC per eseguire il backup dei dati su un nastro virtuale. Ad esempio, è possibile utilizzare istantanee UFS o ZFS.	Sì Questa opzione di backup richiede uno script automatico per cancellare le transazioni in corso, configurare una modalità offline temporanea o di sola lettura durante l'istantanea del file system o eseguire il dump dei dati dell'applicazione. È inoltre possibile richiedere l'inattività delle applicazioni o la modalità di sola lettura.	Sì	Backup dei file system del server Sun SPARC con file.tar o.zip Backup dei dati delle applicazioni
Opzione 2: istantanea di Stromasys	Sì È necessario configurare Charon-SSP Manager o utilizzare un argomento di avvio della riga	Sì Questa opzione di backup crea un'istantanea del server guest emulato, inclusi i dischi virtuali	No	Istantanea del server Sun SPARC Backup dei dati delle applicazioni

di comando per abilitare questa funzionalità.

È inoltre necessario eseguire un comando Linux per chiedere all'emulatore Charon-SSP di salvare lo stato del server guest Sun SPARC in un file snapshot.

> **Important**
> È necessario spegnere il server guest Sun SPARC.

e il dump di memoria.

> **Important**
> È necessario spegnere il server guest Sun SPARC durante l'istantanea.

Opzione 3: istantanea del volume Amazon EBS	Sì Puoi utilizzarlo AWS Backup per automatizzare lo snapshot di Amazon EBS.	Sì Questa opzione di backup richiede uno script automatico per scaricare le transazioni in corso e configurare un arresto temporaneo o di sola lettura dell'istanza Amazon EC2 durante lo snapshot del volume Amazon EBS. ⚠ Important Questa opzione di backup potrebbe richiedere tempi di inattività delle applicazioni o la modalità di sola lettura per garantire	No	Istantanea dei file system del server Sun SPARC Backup dei dati delle applicazioni

la coerenza dell'appl icazione.

Opzione 4: Gateway di archiviazione AWS VTL	Sì È possibile eseguire automaticamente il backup dei dati di backup del file system Sun SPARC sul VTL utilizzando un agente di backup.	Sì Questa opzione di backup richiede uno script automatico per cancellare le transazioni in corso e configurare una modalità offline temporanea o di sola lettura durante l'istantanea del file system o il dump dei dati dell'applicazione. **Important** Questa opzione di backup potrebbe richiedere tempi di inattività delle applicazioni o la modalità di sola lettura.	Sì	Un'ampia gamma di backup dei file system del server Sun SPARC Backup dei dati delle applicazioni

Limitazioni

- È possibile utilizzare gli approcci di questo modello per eseguire il backup di singoli server Sun SPARC, ma è anche possibile utilizzare queste opzioni di backup per dati condivisi se si dispone di applicazioni eseguite in un cluster.

Tools (Strumenti)

Opzione di backup 1: nastro virtuale Stromasys

- L'emulatore Stromasys Charon-SSP crea la replica virtuale dell'hardware SPARC originale all'interno di un sistema informatico standard compatibile con x86 a 64 bit. Esegue il codice binario SPARC originale, inclusi i sistemi operativi (OSs) come SunOS o Solaris, i relativi prodotti a più livelli e le applicazioni.
- Amazon Elastic Compute Cloud (Amazon EC2) Elastic Compute Cloud (Amazon EC2) è un servizio Web che fornisce capacità di calcolo ridimensionabile da utilizzare per creare e ospitare i sistemi software.
- Amazon Elastic File System (Amazon EFS) fornisce un file system semplice, senza server set-and-forget ed elastico da utilizzare con Servizi AWS risorse locali.
- Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è uno storage per Internet.
- AWS Systems Manager L'automazione semplifica le attività comuni di manutenzione e distribuzione delle istanze Amazon EC2 e di altre risorse. AWS

Opzione di backup 2: istantanea Stromasys

- L'emulatore Stromasys Charon-SSP crea la replica virtuale dell'hardware SPARC originale all'interno di un sistema informatico standard compatibile con x86 a 64 bit. Esegue il codice binario SPARC originale, inclusi SunOS o Solaris, i loro prodotti e OSs applicazioni a più livelli.
- Amazon Elastic Compute Cloud (Amazon EC2) Elastic Compute Cloud (Amazon EC2) è un servizio Web che fornisce capacità di calcolo ridimensionabile da utilizzare per creare e ospitare i sistemi software.
- Amazon Elastic File System (Amazon EFS) fornisce un file system semplice, senza server set-and-forget ed elastico da utilizzare con Servizi AWS risorse locali.

- Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è uno storage per Internet.
- AWS Systems Manager L'automazione semplifica le attività comuni di manutenzione e distribuzione delle istanze Amazon EC2 e di altre risorse. AWS

Opzione di backup 3: istantanea del volume Amazon EBS

- L'emulatore di emulatore Stromasys Charon-SSP crea la replica virtuale dell'hardware SPARC originale all'interno di un sistema informatico standard a 64 bit compatibile con x86. Esegue il codice binario SPARC originale, inclusi SunOS o Solaris, i loro prodotti e OSs applicazioni a più livelli.
- AWS Backupè un servizio di protezione dei dati completamente gestito che semplifica la centralizzazione e l'automazione su Servizi AWS, nel cloud e in locale.
- Amazon Elastic Block Store (Amazon EBS) fornisce volumi di storage a livello di blocco da utilizzare con le istanze Amazon EC2.
- Amazon Elastic Compute Cloud (Amazon EC2) Elastic Compute Cloud (Amazon EC2) è un servizio Web che fornisce capacità di calcolo ridimensionabile da utilizzare per creare e ospitare i sistemi software.

Opzione di backup 4: Gateway di archiviazione AWSVTL

- L'emulatore Stromasys Charon-SSP crea la replica virtuale dell'hardware SPARC originale all'interno di un sistema informatico standard compatibile con x86 a 64 bit. Esegue il codice binario SPARC originale, inclusi SunOS o Solaris, i loro prodotti e OSs applicazioni a più livelli.
- Bacula è un sistema di backup informatico open source di livello aziendale. Per ulteriori informazioni sul fatto che l'applicazione di backup esistente supporti Tape Gateway, consulta Applicazioni di backup di terze parti supportate per un Tape Gateway nella documentazione. Gateway di archiviazione AWS
- Amazon Elastic Compute Cloud (Amazon EC2) Elastic Compute Cloud (Amazon EC2) è un servizio Web che fornisce capacità di calcolo ridimensionabile da utilizzare per creare e ospitare i sistemi software.

- Amazon Relational Database Service (Amazon RDS) per MySQL supporta istanze DB che eseguono diverse versioni di MySQL.
- Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è uno storage per Internet.
- Gateway di archiviazione AWScollega un'appliance software locale con uno storage basato sul cloud per fornire una perfetta integrazione con le funzionalità di sicurezza dei dati tra l'ambiente IT locale e l'infrastruttura di storage. AWS

Epiche

Opzione di backup 1: creare un backup su nastro virtuale Stromasys

Operazione	Description	Competenze richieste
Crea un file system condiviso Amazon EFS per lo storage di file su nastro virtuale.	Accedi Console di gestione AWS o usa AWS Command Line Interface (AWS CLI) per creare un file system Amazon EFS. Per ulteriori informazioni su questo argomento, consulta Creare un file system Amazon EFS nella documentazione di Amazon EFS.	Architetto del cloud
Configura l'host Linux per montare il file system condiviso.	Installa il driver Amazon EFS sull'istanza Amazon EC2 Linux e configura il sistema operativo Linux per montare il file system condiviso Amazon EFS durante l'avvio. Per ulteriori informazioni su questo argomento, consulta Mounting file system using Amazon EFS mount helper	DevOps ingegnere

Operazione	Description	Competenze richieste
	nella documentazione di Amazon EFS.	
Installa l'emulatore Charon-SSP.	Installa l'emulatore Charon-SSP sull'istanza Linux di Amazon EC2. Per ulteriori informazioni su questo argomento, consulta Configurazione di un' Cloud AWS istanza per Charon-SSP nella documentazione di Stromasys.	DevOps ingegnere
Crea un contenitore di file su nastro virtuale nel file system condiviso per ogni server guest Sun SPARC.	Esegui il `touch <vtape-container-name>` comando per creare un contenitore di file su nastro virtuale nel file system condiviso per ogni server guest Sun SPARC distribuito nell'emulatore Charon-SSP.	DevOps ingegnere
Configura Charon-SSP Manager per creare dispositivi a nastro virtuali per i server guest Sun SPARC.	Accedere a Charon-SSP Manager, creare dispositivi a nastro virtuali e configurarli per utilizzare i file contenitore dei nastri virtuali per ogni server guest Sun SPARC. Per ulteriori informazioni su questo argomento, consultate la guida per l'utente di Charon-SSP 5.2 per Linux nella documentazione di Stromasys.	DevOps ingegnere

Operazione	Description	Competenze richieste
Verificare che il dispositivo a nastro virtuale sia disponibile nei server guest Sun SPARC.	Accedere a ciascun server guest Sun SPARC ed eseguire il `mt -f /dev/rmt/1` comando per verificare che il dispositivo a nastro virtuale sia configurato nel sistema operativo.	DevOps ingegnere
Sviluppa il runbook e l'automazione di Systems Manager Automation.	Sviluppa il runbook Systems Manager Automation e configura finestre e associazioni di manutenzione in Systems Manager per pianificare il processo di backup. Per ulteriori informazioni su questo argomento, vedere Procedure dettagliate sull'automazione e Configurazione delle finestre di manutenzione nella documentazione. AWS Systems Manager	Architetto del cloud
Configurare Systems Manager Automation per archiviare i file contenitori di nastri virtuali ruotati.	Usa l'esempio di codice dell'opzione Back 1 nella sezione Informazioni aggiuntive per sviluppare un runbook di Systems Manager Automation per archiviare i file container di nastri virtuali ruotati su Amazon S3.	Architetto del cloud

Operazione	Description	Competenze richieste
Implementa il runbook Systems Manager Automatio n per l'archiviazione e la pianificazione.	Implementa il runbook Systems Manager Automatio n e pianificane l'esecuzi one automatica in Systems Manager. Per ulteriori informazioni su questo argomento, vedere le procedure dettagliate di automazione nella documenta zione di Systems Manager.	Architetto del cloud

Opzione di backup 2: creare un'istantanea di Stromasys

Operazione	Description	Competenze richieste
Crea un file system condiviso Amazon EFS per lo storage di file su nastro virtuale.	Accedi a Console di gestione AWS o usalo AWS CLI per creare un file system Amazon EFS. Per ulteriori informazioni su questo argomento, consulta Crea il tuo file system Amazon EFS nella documentazione di Amazon EFS.	Architetto del cloud
Configura l'host Linux per montare il file system condiviso.	Installa il driver Amazon EFS nell'istanza Amazon EC2 Linux e configura il sistema operativo Linux per montare il file system condiviso Amazon EFS durante l'avvio.	DevOps ingegnere

Operazione	Description	Competenze richieste
	Per ulteriori informazioni su questo argomento, consulta Mounting file system using Amazon EFS mount helper nella documentazione di Amazon EFS.	
Installa l'emulatore Charon-SSP.	Installa l'emulatore Charon-SSP sull'istanza Linux di Amazon EC2. Per ulteriori informazioni su questo argomento, consulta Configurazione di un' Cloud AWS istanza per Charon-SSP nella documentazione di Stromasys.	DevOps ingegnere
Configura i server guest Sun SPARC per l'avvio con l'opzione snapshot.	Usa Charon-SSP Manager per configurare l'opzione snapshot per ogni server guest Sun SPARC. Per ulteriori informazioni su questo argomento, consultate la guida per l'utente di Charon-SSP 5.2 per Linux nella documentazione di Stromasys.	DevOps ingegnere

Operazione	Description	Competenze richieste
Sviluppa il runbook Systems Manager Automation.	Utilizzare l'esempio di codice dell'opzione Backup 2 nella sezione Informazioni aggiuntive per sviluppare un runbook Systems Manager Automation per eseguire in remoto il comando snapshot su un server guest Sun SPARC durante una finestra di manutenzione.	Architetto del cloud
Implementa il runbook Systems Manager Automation e configura l'associazione agli host Amazon EC2 Linux.	Implementa il runbook Systems Manager Automation e configura finestre e associazioni di manutenzione in Systems Manager per pianificare il processo di backup. Per ulteriori informazioni su questo argomento, consulta le procedure dettagliate sull'automazione e la configurazione delle finestre di manutenzione nella documentazione. AWS Systems Manager	Architetto del cloud
Archivia le istantanee in uno storage a lungo termine.	Usa il codice di esempio del runbook dalla sezione Informazioni aggiuntive per sviluppare un runbook Systems Manager Automation per archiviare i file di snapshot su Amazon S3.	Architetto del cloud

Opzione di backup 3: creazione di uno snapshot del volume Amazon EBS

Operazione	Description	Competenze richieste
Installa l'emulatore Charon-SSP.	Installa l'emulatore Charon-SSP sull'istanza Linux di Amazon EC2. Per ulteriori informazioni su questo argomento, consulta Configurazione di un' Cloud AWS istanza per Charon-SSP nella documentazione di Stromasys.	DevOps ingegnere
Crea volumi Amazon EBS per i server guest Sun SPRAC.	Accedi a Console di gestione AWS, apri la console Amazon EBS e crea volumi Amazon EBS per i server guest Sun SPRAC. Per ulteriori informazioni su questo argomento, consulta Configurazione di un' Cloud AWS istanza per Charon-SSP nella documentazione di Stromasys.	Architetto del cloud
Collega i volumi Amazon EBS all'istanza Amazon EC2 Linux.	Sulla console Amazon EC2, collega i volumi Amazon EBS all'istanza Amazon EC2 Linux. Per ulteriori informazioni su questo argomento, consulta Collegare un volume Amazon EBS a un'istanza nella documentazione di Amazon EC2.	AWS DevOps

Operazione	Description	Competenze richieste
Mappa i volumi Amazon EBS come unità SCSI nell'emul atore Charon-SSP.	Configura Charon-SSP Manager per mappare i volumi Amazon EBS come unità SCSI nei server guest Sun SPARC. Per ulteriori informazioni su questo argomento, consulta la sezione sulla configura zione dello storage SCSI della guida Charon-SSP V5.2 per Linux nella documentazione di Stromasys.	AWS DevOps
Configura la AWS Backup pianificazione per lo snapshot dei volumi Amazon EBS.	Configura AWS Backup policy e pianificazioni per eseguire lo snapshot dei volumi Amazon EBS. Per ulteriori informazioni su questo argomento, consulta il AWS Backup tutorial sull'utilizzo di backup e ripristin o di Amazon EBS nella documentazione del AWS Developer Center.	AWS DevOps

Opzione di backup 4: creazione di un Gateway di archiviazione AWS VTL

Operazione	Description	Competenze richieste
Crea un dispositivo Tape Gateway.	Accedi a Console di gestione AWS, apri la Gateway di archiviazione AWS console	Architetto del cloud

Operazione	Description	Competenze richieste
	e crea un dispositivo Tape Gateway in un VPC. Per ulteriori informazioni su questo argomento, consulta Creazione di un gateway nella Gateway di archiviazione AWS documentazione.	
Crea un'istanza database Amazon RDS per il catalogo Bacula.	Apri la console Amazon RDS e crea un'istanza database Amazon RDS for MySQL. Per ulteriori informazioni su questo argomento, consulta Creazione di un'istanza DB MySQL e connessione a un database su un'istanza DB MySQL nella documentazione di Amazon RDS.	Architetto del cloud

Operazione	Description	Competenze richieste
Implementa il controller dell'applicazione di backup nel VPC.	Installa Bacula sull'istanza Amazon EC2, distribuisci il controller dell'applicazione di backup, quindi configura lo storage di backup per la connessione con il dispositivo Tape Gateway. È possibile utilizzare la configurazione di esempio dello storage daemon di Bacula Director contenuta nel file (allegato). `Bacula-storage-daemon-config.txt` Per ulteriori informazioni su questo argomento, consulta la documentazione di Bacula.	AWS DevOps
Configura l'applicazione di backup sui server guest Sun SPARC.	Configurare un secondo client per installare e configurare l'applicazione di backup sui server guest Sun SPARC utilizzando la configurazione Bacula di esempio contenuta nel `SUN-SPARC-Guest-Bacula-Config.txt` file (allegato).	DevOps ingegnere

Operazione	Description	Competenze richieste
Imposta la configurazione e la pianificazione del backup.	Imposta la configurazione e le pianificazioni di backup nel controller dell'applicazione di backup utilizzando la configurazione di esempio di Bacula Director contenuta nel `Bacula-Directory-Config.txt` file (allegato). Per ulteriori informazioni su questo argomento, consulta la documentazione di Bacula.	DevOps ingegnere
Verifica che la configurazione e le pianificazioni di backup siano corrette.	Segui le istruzioni contenute nella documentazione di Bacula per eseguire i test di convalida e backup della configurazione nei server guest Sun SPARC. Ad esempio, è possibile utilizzare i seguenti comandi per convalidare i file di configurazione: • `bacula-dir -t -c bacula-dir.conf` • `bacula-fd -t -c bacula-fd.conf` • `bacula-sd -t -c bacula-sd.conf`	DevOps ingegnere

Risorse correlate

- Charon virtual SPARC con licenza VE

- SPARC virtuale Charon
- Utilizzo dei servizi cloud e dello storage di oggetti con Bacula Enterprise Edition
- Obiettivi di disaster recovery (DR)
- Soluzioni di emulazione del sistema Charon Legacy

Informazioni aggiuntive

Opzione di backup 1: creazione di un nastro virtuale Stromasys

È possibile utilizzare il seguente codice di runbook di esempio di Systems Manager Automation per avviare automaticamente il backup e quindi scambiare i nastri:

```
...
# example backup script saved in SUN SPARC Server
 #!/usr/bin/bash
 mt -f  rewind
 tar -cvf
 mt -f  offline
...
         mainSteps:
         - action: aws:runShellScript
           name:
           inputs:
             onFailure: Abort
             timeoutSeconds: "1200"
             runCommand:
             - |
               # Validate tape backup container file exists
               if [ ! -f {{TapeBackupContainerFile}} ]; then
                 logger -s -p local3.warning "Tape backup container file is not exists
 - {{TapeBackupContainerFile}}, create a new one"
                 touch {{TapeBackupContainerFile}}
               fi
         - action: aws:runShellScript
           name: startBackup
           inputs:
             onFailure: Abort
             timeoutSeconds: "1200"
             runCommand:
             - |
               user={{BACKUP_USER}}
```

```
              keypair={{KEYPAIR_PATH}}
              server={{SUN_SPARC_IP}}
              backup_script={{BACKUP_SCRIPT}}
              ssh -i $keypair $user@$server -c "/usr/bin/bash $backup_script"
        - action: aws:runShellScript
          name: swapVirtualDiskContainer
          inputs:
            onFailure: Abort
            timeoutSeconds: "1200"
            runCommand:
            - |
              mv {{TapeBackupContainerFile}} {{TapeBackupContainerFile}}.$(date +%s)
              touch {{TapeBackupContainerFile}}
        - action: aws:runShellScript
          name: uploadBackupArchiveToS3
          inputs:
            onFailure: Abort
            timeoutSeconds: "1200"
            runCommand:
            - |
              aws s3 cp {{TapeBackupContainerFile}} s3://{{BACKUP_BUCKET}}/
{{SUN_SPARC_IP}}/$(date '+%Y-%m-%d')/
 ...
```

Opzione di backup 2: istantanea Stromasys

È possibile utilizzare il seguente codice di runbook di esempio di Systems Manager Automation per automatizzare il processo di backup:

```
      ...

         mainSteps:
         - action: aws:runShellScript
           name: startSnapshot
           inputs:
             onFailure: Abort
             timeoutSeconds: "1200"
             runCommand:
             - |
               # You may consider some graceful stop of the application before taking a
 snapshot
               # Query SSP PID by configuration file
```

```
              # Example: ps ax | grep ssp-4 | grep Solaris10.cfg | awk '{print $1"
 "$5}' | grep ssp4 | cut -f1 -d" "
              pid=`ps ax | grep ssp-4 | grep {{SSP_GUEST_CONFIG_FILE}} | awk '{print
 $1" "$5}' | grep ssp4 | cut -f1 -d" "`
              if [ -n "${pid}" ]; then
                kill -SIGTSTP ${pid}
              else
                echo "No PID found for SPARC guest with config
 {{SSP_GUEST_CONFIG_FILE}}"
                exit 1
              fi
        - action: aws:runShellScript
          name: startBackup
          inputs:
            onFailure: Abort
            timeoutSeconds: "1200"
            runCommand:
            - |
              # upload snapshot and virtual disk files into S3
              aws s3 sync {{SNAPSHOT_FOLDER}} s3://{{BACKUP_BUCKET}}/$(date '+%Y-%m-
%d')/
              aws s3 cp {{VIRTUAL_DISK_FILE}} s3://{{BACKUP_BUCKET}}/$(date '+%Y-%m-
%d')/
        - action: aws:runShellScript
          name: restratSPARCGuest
          inputs:
            onFailure: Abort
            timeoutSeconds: "1200"
            runCommand:
            - |
              /opt/charon-ssp/ssp-4u/ssp4u -f {{SSP_GUEST_CONFIG_FILE}} -d -a
 {{SPARC_GUEST_NAME}} --snapshot {{SNAPSHOT_FOLDER}}
 ...
```

Opzione di backup 4 —Gateway di archiviazione AWS VTL

Se si utilizzano zone non globali di Solaris per eseguire server Sun SPARC legacy virtualizzati, l'approccio dell'applicazione di backup può essere applicato alle zone non globali in esecuzione sui server Sun SPARC (ad esempio, il client di backup può essere eseguito all'interno delle zone non globali). Tuttavia, il client di backup può essere eseguito anche sull'host Solaris e scattare istantanee delle zone non globali. Le istantanee possono quindi essere salvate su nastro.

La seguente configurazione di esempio aggiunge il file system che ospita le zone non globali di Solaris alla configurazione di backup per l'host Solaris:

```
FileSet {
   Name = "Branded Zones"
   Include {
     Options {
       signature = MD5
     }
     File = /zones
   }
 }
```

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Esegui il backup e l'archiviazione dei dati su Amazon S3 con Veeam Backup & Replication

Jeanna James, Anthony Fiore (AWS) e William Quigley, Amazon Web Services

Riepilogo

Questo modello descrive in dettaglio il processo di invio dei backup creati da Veeam Backup & Replication alle classi di storage di oggetti Amazon Simple Storage Service (Amazon S3) supportate utilizzando la funzionalità di repository di backup scale-out di Veeam.

Veeam supporta più classi di storage Amazon S3 per soddisfare al meglio le tue esigenze specifiche. Puoi scegliere il tipo di storage in base all'accesso ai dati, alla resilienza e ai requisiti di costo dei tuoi dati di backup o archiviazione. Ad esempio, puoi archiviare dati che non prevedi di utilizzare per 30 giorni o più in Amazon S3 Infrequent Access (IA) a un costo inferiore. Se hai intenzione di archiviare i dati per 90 giorni o più, puoi utilizzare S3 Glacier Flexible Retrieval o S3 Glacier Deep Archive con il livello di archiviazione di Veeam. Puoi anche utilizzare S3 Object Lock per creare backup immutabili all'interno di Amazon S3.

Questo modello non illustra come configurare Veeam Backup & Replication con un gateway a nastro. Gateway di archiviazione AWS Per informazioni su questo argomento, consulta Veeam Backup & Replication using AWS VTL Gateway - Deployment Guide sul sito Web Veeam.

> Avvertenza: questo scenario richiede utenti AWS Identity and Access Management (IAM) con accesso programmatico e credenziali a lungo termine, che presentano un rischio per la sicurezza . Per ridurre questo rischio, si consiglia di fornire a questi utenti solo le autorizzazioni necessarie per eseguire l'attività e di rimuoverli quando non sono più necessari. Le chiavi di accesso possono essere aggiornate se necessario. Per ulteriori informazioni, consulta Aggiornamento delle chiavi di accesso nella Guida per l'utente IAM.

Prerequisiti e limitazioni

Prerequisiti

- Veeam Backup & Replication, incluso Veeam Availability Suite o Veeam Backup Essentials, installato (puoi registrarti per una prova gratuita)
- Licenza Veeam Backup & Replication con funzionalità Enterprise o Enterprise Plus, che include la Veeam Universal License (VUL)

- Un utente IAM attivo con accesso a un bucket Amazon S3
- Un utente IAM attivo con accesso ad Amazon Elastic Compute Cloud (Amazon EC2) e Amazon Virtual Private Cloud (Amazon VPC), se si utilizza il livello di archiviazione
- Connettività di rete da locale a larghezza Servizi AWS di banda disponibile per il backup e il ripristino del traffico tramite una connessione Internet pubblica o un'interfaccia virtuale AWS Direct Connect pubblica (VIF)
- Le seguenti porte ed endpoint di rete sono stati aperti per garantire una comunicazione corretta con gli archivi di storage di oggetti:
  - Storage Amazon S3 — TCP — porta 443: utilizzata per comunicare con lo storage Amazon S3.
  - Storage Amazon S3 — endpoint cloud — `*.amazonaws.com` per Regioni AWS e per le AWS GovCloud (US) Regions regioni della `*.amazonaws.com.cn` Cina: utilizzato per comunicare con lo storage Amazon S3. Per un elenco completo degli endpoint di connessione, consulta gli endpoint Amazon S3 nella documentazione. AWS
  - Storage Amazon S3 — TCP HTTP — porta 80: utilizzata per verificare lo stato del certificato. Tieni presente che gli endpoint di verifica dei certificati, ovvero i server CRL (Certificate Revocation List) URLs e Online Certificate Status Protocol (OCSP), sono soggetti a modifiche. L'elenco effettivo degli indirizzi si trova nel certificato stesso.
  - Storage Amazon S3 — endpoint di verifica dei certificati —`*.amazontrust.com`: utilizzato per verificare lo stato del certificato. Tieni presente che gli endpoint di verifica dei certificati (server CRL URLs e OCSP) sono soggetti a modifiche. L'elenco effettivo degli indirizzi si trova nel certificato stesso.

Limitazioni

- Veeam non supporta le policy S3 Lifecycle su nessun bucket S3 utilizzato come repository di storage di oggetti Veeam. Queste includono policy con transizioni di classi di storage Amazon S3 e regole di scadenza del ciclo di vita di Amazon S3. Veeam deve essere l'unica entità che gestisce questi oggetti. L'attivazione delle policy del ciclo di vita di S3 potrebbe avere risultati imprevisti, inclusa la perdita di dati.

Versioni del prodotto

- Veeam Backup & Replication v9.5 Update 4 o successivo (solo backup o livello di capacità)
- Veeam Backup & Replication v10 o successivo (backup o livello di capacità e S3 Object Lock)

- Veeam Backup & Replication v11 o successivo (livello di backup o capacità, livello di archiviazione o archiviazione e S3 Object Lock)
- Veeam Backup & Replication v12 o successivo (livello di prestazioni, livello di backup o capacità, livello di archiviazione o livello di archiviazione e S3 Object Lock)
- S3 Standard
- S3 Standard-IA
- S3 One Zone-IA
- S3 Glacier Flexible Retrieval (solo v11 e versioni successive)
- S3 Glacier Deep Archive (solo v11 e successive)
- S3 Glacier Instant Retrieval (solo v12 e successive)

Architecture

Stack tecnologico di origine

- Installazione locale di Veeam Backup & Replication con connettività da un server di backup Veeam o da un server gateway Veeam ad Amazon S3

Stack tecnologico Target

- Simple Storage Service (Amazon S3)
- Amazon VPC e Amazon EC2 (se si utilizza il livello di archiviazione)

Architettura di destinazione: SOBR

Il diagramma seguente mostra l'architettura Scale-out Backup Repository (SOBR).

Il software Veeam Backup and Replication protegge i dati da errori logici come guasti del sistema, errori delle applicazioni o cancellazioni accidentali. In questo diagramma, i backup vengono eseguiti prima in locale e una copia secondaria viene inviata direttamente ad Amazon S3. Un backup rappresenta una point-in-time copia dei dati.

Il flusso di lavoro è composto da tre componenti principali necessari per la suddivisione in più livelli o la copia dei backup su Amazon S3 e un componente opzionale:

- Veeam Backup & Replication (1) — Il server di backup responsabile del coordinamento, del controllo e della gestione dell'infrastruttura di backup, delle impostazioni, dei job, delle attività di ripristino e di altri processi.
- Server gateway Veeam (non mostrato nel diagramma): un server gateway locale opzionale necessario se il server di backup Veeam non dispone di connettività in uscita ad Amazon S3.
- Repository di backup scalabile (2): sistema di repository con supporto per la scalabilità orizzontale per lo storage dei dati a più livelli. L'archivio di backup con scalabilità orizzontale è costituito da uno o più repository di backup che forniscono un accesso rapido ai dati e può essere ampliato con i repository di storage di oggetti Amazon S3 per lo storage a lungo termine (livello di capacità) e l'archiviazione (livello di archiviazione). Veeam utilizza il repository di backup scalabile per suddividere automaticamente i dati tra lo storage di oggetti locale (livello di prestazioni) e lo storage di oggetti Amazon S3 (livelli di capacità e archiviazione).

  > **Note**
  >
  > A partire da Veeam Backup & Replication v12.2, la funzionalità Direct to S3 Glacier rende opzionale il livello di capacità S3. Un SOBR può essere configurato con un livello di prestazioni e un livello di archiviazione S3 Glacier. Questa configurazione è utile per gli utenti che hanno investimenti significativi nello storage locale (on-premise) per il livello di capacità e che richiedono solo la conservazione a lungo termine degli archivi nel cloud. Per ulteriori informazioni, consulta la documentazione di Veeam Backup & Replication.

- Amazon S3 (3): servizio di storage di AWS oggetti che offre scalabilità, disponibilità dei dati, sicurezza e prestazioni.

Architettura di destinazione: DTO

Il diagramma seguente mostra l'architettura direct-to-object (DTO).

In questo diagramma, i dati di backup vengono trasferiti direttamente ad Amazon S3 senza essere prima archiviati in locale. Le copie secondarie possono essere archiviate in S3 Glacier.

Automazione e scalabilità

Puoi automatizzare la creazione di risorse IAM e bucket S3 utilizzando i CloudFormation modelli forniti nel repository. VeeamHub GitHub I modelli includono opzioni standard e immutabili.

Tools (Strumenti)

Strumenti e Servizi AWS

- Veeam Backup & Replication è una soluzione di Veeam per la protezione, il backup, la replica e il ripristino dei carichi di lavoro virtuali e fisici.
- CloudFormationti aiuta a modellare e configurare le tue AWS risorse, a fornirle in modo rapido e coerente e a gestirle durante tutto il loro ciclo di vita. È possibile utilizzare un modello per descrivere le risorse e le relative dipendenze e lanciarle e configurarle insieme come uno stack, anziché gestire le risorse singolarmente. Puoi gestire ed eseguire il provisioning degli stack su più sistemi. Account AWS Regioni AWS
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi usare Amazon EC2 per lanciare tutti o pochi server virtuali di cui hai bisogno, con scalabilità orizzontale o orizzontale.
- AWS Identity and Access Management (IAM) è un servizio web per il controllo sicuro degli accessi a Servizi AWS. Con IAM, puoi gestire centralmente gli utenti, le credenziali di sicurezza come le chiavi di accesso e le autorizzazioni che controllano a quali AWS risorse possono accedere utenti e applicazioni.
- Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è un servizio di storage di oggetti. È possibile utilizzare Amazon S3 per memorizzare e recuperare qualsiasi volume di dati, in qualunque momento e da qualunque luogo tramite il Web.
- Amazon S3 Glacier (S3 Glacier) è un servizio sicuro e durevole per l'archiviazione dei dati a basso costo e il backup a lungo termine.
- Amazon Virtual Private Cloud (Amazon VPC) fornisce una sezione logicamente isolata Cloud AWS in cui è possibile avviare AWS le risorse in una rete virtuale definita dall'utente. Questa rete virtuale è simile a una comune rete da gestire all’interno del proprio data center, ma con i vantaggi dell’infrastruttura scalabile di AWS.

Codice

Utilizza i CloudFormation modelli forniti nel VeeamHub GitHub repository per creare automaticamente le risorse IAM e i bucket S3 per questo modello. Se preferisci creare queste risorse manualmente, segui i passaggi nella sezione Epics.

Best practice

In conformità con le best practice IAM, ti consigliamo vivamente di ruotare regolarmente le credenziali utente IAM a lungo termine, come l'utente IAM che utilizzi per scrivere i backup di Veeam Backup & Replication su Amazon S3. Per ulteriori informazioni, consulta Best practice di sicurezza nella documentazione di IAM.

Epiche

Configura lo storage Amazon S3 nel tuo account

Operazione	Description	Competenze richieste
Crea un utente IAM.	Segui le istruzioni nella documentazione IAM per creare un utente IAM. Questo utente non dovrebbe avere accesso alla AWS console e dovrai creare una chiave di accesso per questo utente. Veeam utilizza questa entità per l'autenticazione, la lettura e AWS la scrittura nei bucket S3. È necessario concedere il privilegio minimo (ovvero concedere solo le autorizza zioni necessarie per eseguire un'attività) in modo che l'utente non abbia più autorità di quella necessaria. Ad esempio, le policy IAM da collegare al tuo utente Veeam IAM, consulta la sezione Informazioni aggiuntive.	Amministratore AWS

Operazione	Description	Competenze richieste
	Note In alternativa, puoi utilizzare i CloudFormation modelli forniti nel VeeamHub GitHub repository per creare un utente IAM e un bucket S3 per questo modello.	

Operazione	Description	Competenze richieste
Crea un bucket S3.	1. Accedi Console di gestione AWS e apri la console Amazon S3. 2. Se non disponi già di un bucket S3 esistente da utilizzare come storage di destinazione, scegli Crea bucket e specifica il nome del bucket e le impostazioni del bucket. Regione AWS • Ti consigliamo di abilitare l'opzione Block Public Access per il bucket S3 e di configurare le politiche di accesso e autorizza zione degli utenti per soddisfare i requisiti della tua organizzazione. Per un esempio, consulta la documentazione di Amazon S3. • Ti consigliamo di abilitare S3 Object Lock, anche se non intendi utilizzarlo subito. Questa impostazi one può essere abilitata solo al momento della creazione del bucket S3. Per ulteriori informazioni, consulta Creazione di un bucket nella documentazione di Amazon S3.	Amministratore AWS

Aggiungi Amazon S3 e S3 Glacier Flexible Retrieval (o S3 Glacier Deep Archive) a Veeam Backup & Replication

Operazione	Description	Competenze richieste
Avvia la procedura guidata New Object Repository.	Prima di configurare lo storage di oggetti e gli archivi di backup con scalabilità orizzontale in Veeam, devi aggiungere i repository di storage Amazon S3 e S3 Glacier che desideri utilizzar e per i livelli di capacità e archiviazione. Nella prossima epopea, collegherai questi repository di storage al tuo repository di backup scalabile. 1. Sulla console Veeam, apri la vista dell'infrastruttura di Backup. 2. Nel riquadro dell'inventario, scegli il nodo Backup Repository, quindi scegli Aggiungi repository. 3. Nella finestra di dialogo Aggiungi archivio di backup, scegli Object Storage, Amazon S3.	Amministratore AWS, proprietario dell'app
Aggiungi lo storage Amazon S3 per il livello di capacità.	1. Nella finestra di dialogo Amazon Cloud Storage Services, scegli Amazon S3. 2. Nella fase Nome della procedura guidata, specifica il nome di archiviazione	Amministratore AWS, proprietario dell'app

Operazione	Description	Competenze richieste
	dell'oggetto e una breve descrizione, ad esempio il creatore e la data di creazione. 3. Nella fase Account della procedura guidata, specificare l'account di archiviazione degli oggetti. • Per le credenziali, scegli l'utente IAM che hai creato nella prima epic per accedere allo storage di oggetti Amazon S3. • Per la regione AWS, scegli Regione AWS dove si trova il bucket S3. 4. Nella fase Bucket della procedura guidata, specifica le impostazioni di archiviazione degli oggetti. • Per la regione del data center, scegli Regione AWS dove si trova il bucket S3. • Per Bucket, scegli il bucket S3 che hai creato nella prima epic. • Per Cartella, crea o seleziona una cartella cloud su cui mappare il tuo repository di archiviazione degli oggetti.	

Operazione	Description	Competenze richieste
	• Se desideri abilitare l'immutabilità, scegli Rendi i backup recenti immutabili per X giorni e imposta il periodo di tempo durante il quale i backup devono essere bloccati. Tieni presente che l'abilita zione dell'immutabilità comporta un aumento dei costi a causa dell'aume nto del numero di chiamate API ad Amazon S3 da Veeam. 5. Nella fase di riepilogo della procedura guidata, rivedi le informazioni di configura zione, quindi scegli Fine.	

Operazione	Description	Competenze richieste
Aggiungi lo storage S3 Glacier per il livello di archiviazione.	Se desideri creare un livello di archiviazione, utilizza le autorizzazioni IAM dettagliate nella sezione Informazioni aggiuntive. 1. Avvia la procedura guidata New Object Repository come descritto in precedenza. 2. Nella finestra di dialogo Amazon Cloud Storage Services, scegli Amazon S3 Glacier. 3. Nella fase Nome della procedura guidata, specifica il nome di archiviazione dell'oggetto e una breve descrizione, ad esempio il creatore e la data di creazione. 4. Nella fase Account della procedura guidata, specificare l'account di archiviazione degli oggetti. • Per le credenziali, scegli l'utente IAM che hai creato nella prima epic per accedere allo storage di oggetti S3 Glacier. • Per la regione AWS, scegli Regione AWS dove si trova il bucket S3.	Amministratore AWS, proprietario dell'app

Operazione	Description	Competenze richieste
	5. Nella fase Bucket della procedura guidata, specifica le impostazioni di archiviazione degli oggetti. • Per la regione del centro dati, scegli. Regione AWS • Per Bucket, scegli un bucket S3 per archiviare i dati di backup. Può trattarsi dello stesso bucket utilizzato per il livello di capacità. • Per Cartella, crea o seleziona una cartella cloud su cui mappare il tuo repository di archiviazione degli oggetti. • Se desideri abilitare l'immutabilità, scegli Rendi immutabili i backup recenti per l'intera durata della loro politica di conservazione. Tieni presente che l'abilitazione dell'immutabilità comporta un aumento dei costi a causa dell'aumento del numero di chiamate API ad Amazon S3 da Veeam. • Se desideri utilizzare S3 Glacier Deep Archive come classe di archiviaz	

Operazione	Description	Competenze richieste
	ione, scegli Usa la classe Deep Archive Storage. 6. Nella fase Proxy Appliance della procedura guidata, configura l'istanza ausiliaria utilizzata per trasferire i dati da Amazon S3 a S3 Glacier. È possibile utilizzare le impostazioni predefinite o configurare ogni impostazione manualmente. Per configurare le impostazioni manualmente: • Scegliere Customize (Personalizza). • Ad EC2 esempio, scegliete il tipo di istanza per l'appliance proxy, in base ai requisiti di velocità e costo richiesti per il trasferimento dei file di backup al livello di archiviazione del vostro repository di backup scalabile. • Per Amazon VPC, scegli il VPC per l'istanza di destinazione. • Per Subnet, scegli la sottorete per l'appliance proxy. • Per Gruppo di sicurezza, scegliere il gruppo di	

Operazione	Description	Competenze richieste
	sicurezza da associare all'appliance proxy. • Per la porta Redirecto r, specificare la porta TCP per il routing delle richieste tra l'applian ce proxy e i component i dell'infrastruttura di backup. • Scegli OK per confermare le impostazioni. 7. Nella fase di riepilogo della procedura guidata, rivedi le informazioni di configura zione, quindi scegli Fine.	

Aggiungi repository di backup scalabili

Operazione	Description	Competenze richieste
Avvia la procedura guidata New Scale-Out Backup Repository.	1. Sulla console Veeam, apri la vista dell'infrastruttura di Backup. 2. Nel riquadro dell'inventario, scegli Scale-out Repositor ies, quindi scegli Aggiungi Scale-out Repository.	Proprietario dell'app, amministratore di sistema AWS
Aggiungi un repository di backup con scalabilità orizzontale e configura i livelli di capacità e archiviazione.	1. Nella fase Nome della procedura guidata, specifica il nome e una breve descrizione del repositor y di backup con scalabilità orizzontale.	Proprietario dell'app, amministratore di sistema AWS

Operazione	Description	Competenze richieste
	2. Se necessario, aggiungi estensioni di prestazioni. Puoi anche utilizzare il tuo repository di backup locale Veeam esistente come livello di prestazioni. A partire dalla versione 12 di Veeam, puoi aggiungere un bucket S3 come estension e delle prestazioni per i backup direct-to-object (DTO), bypassando un livello di prestazioni locale. 3. Scegli Advanced e specifica opzioni aggiuntive per il repository di backup scalabile. • Scegli Usa file di backup per computer per creare un file di backup separato per ogni macchina e scrivi questi file nell'arch ivio di backup in più flussi contemporaneamente . Questa opzione è consigliata per un migliore utilizzo delle risorse di archiviazione e di calcolo. • Scegli Esegui backup completo quando l'estensione richiesta è offline per creare un file di backup completo nel	

Operazione	Description	Competenze richieste
	caso in cui un'estens ione che contiene punti di ripristino per un backup incrementale vada offline. Questa opzione richiede spazio libero nell'archivio di backup scalabile per ospitare un file di backup completo. 4. Nella fase relativa alle regole della procedura guidata, specificare la politica di posizionamento dei backup per il repository. • Scegli Data locality per archiviare insieme i file di backup completi e incrementali che appartengono alla stessa catena, con le stesse prestazioni. È possibile archiviare i file che appartengono a una nuova catena di backup con lo stesso livello di prestazioni o con un altro livello di prestazioni (a meno che non si utilizzi un'appliance di archiviaz ione con deduplicazione come livello di prestazio ni). • Scegliete Performan ce per archiviare i file	

Operazione	Description	Competenze richieste
	di backup completi e incrementali con livelli di prestazioni diversi. Questa opzione richiede una connessione di rete veloce e affidabile. Se scegli Prestazioni, puoi limitare i tipi di file di backup da archiviare in base a ciascun livello di prestazioni. Ad esempio, è possibile archiviare file di backup completi su un'estensione e file di backup incrementali su altre estensioni. Per scegliere i tipi di file: • Scegliere Customize (Personalizza). • Nella finestra di dialogo Backup Placement Settings, scegliete un livello di prestazio ni, quindi scegliete Modifica. • Scegliete il tipo di file di backup che desiderat e archiviare nell'este nsione. 5. Nella fase relativa al livello di capacità della procedura guidata, configura il livello di storage a lungo termine da collegare al repository	

Operazione	Description	Competenze richieste
	di backup con scalabilità orizzontale. • Scegli Estendi la capacità del repository di backup con scalabili tà orizzontale con lo storage a oggetti. Per il repository di storage di oggetti, scegli lo storage Amazon S3 per il livello di capacità che hai aggiunto nell'epopea precedente. • Scegli Finestra per selezionare una finestra temporale per lo spostamento o la copia dei dati. • Scegli Copia i backup nell'archiviazione degli oggetti non appena vengono creati per copiare tutti i file di backup creati di recente o solo quelli creati di recente nella misura della capacità. • Scegli Sposta i backup nello storage a oggetti man mano che invecchia no dalla finestra dei ripristini operativi per trasferire le catene di backup inattive al massimo della capacità.	

Operazione	Description	Competenze richieste
	Nel campo Sposta i file di backup più vecchi di X giorni, specifica una durata dopo la quale i file di backup devono essere scaricati. (Per scaricare le catene di backup inattive il giorno in cui sono state create, specifica 0 giorni.) Puoi anche scegliere Override per spostare i file di backup prima se l'archivio di backup con scalabilità orizzontale ha raggiunto una soglia specificata. • Scegliete Crittografa i dati caricati nell'object storage e specificate una password per crittogra fare tutti i dati e i relativi metadati per l'offload. Scegli Aggiungi o gestisci password per specificare una nuova password. 6. Nella fase Archive Tier della procedura guidata, configura il livello di archiviazione che desideri collegare al repository di backup con scalabili tà orizzontale. (Questo passaggio non viene visualizzato se hai saltato	

Operazione	Description	Competenze richieste
	l'aggiunta dello storage Amazon S3 Glacier.) • Scegli Archivia i backup completi GFS sullo storage di oggetti. Per il repository di storage di oggetti, scegli lo storage Amazon Glacier che hai aggiunto nell'epopea precedente. • Per i backup di Archive GFS più vecchi di N giorni, scegli una finestra temporale per spostare i file nell'estensione di archiviazione. (Per archiviare le catene di backup inattive il giorno in cui sono state create, specifica 0 giorni). 7. Nella fase di riepilogo della procedura guidata, esaminate la configurazione del repository di backup con scalabilità orizzontale, quindi scegliete Fine.	

Risorse correlate

- Creazione di un utente IAM nella tua Account AWS (documentazione IAM)
- Creazione di un bucket (documentazione Amazon S3)
- Blocco dell'accesso pubblico allo storage Amazon S3 (documentazione Amazon S3)
- Utilizzo di S3 Object Lock (documentazione di Amazon S3)

- Documentazione tecnica Veeam
- Come creare una policy IAM sicura per la connessione a S3 Object Storage (documentazione Veeam)

Informazioni aggiuntive

Le seguenti sezioni forniscono esempi di policy IAM che è possibile utilizzare quando si crea un utente IAM nella sezione Epics di questo modello.

Policy IAM per il livello di capacità

> **Note**
>
> Cambia il nome dei bucket S3 nella policy di esempio dal `<yourbucketname>` nome del bucket S3 che desideri utilizzare per i backup a livello di capacità di Veeam. Si noti inoltre che la policy deve essere limitata alle risorse specifiche utilizzate per Veeam (indicate dalla `Resource` specifica nella seguente policy) e che la prima parte della policy disabilita la crittografia lato client, come discusso nel post del AWS blog Preventing unintended encryption of Amazon S3 objects.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "RestrictSSECObjectUploads",
            "Effect": "Deny",
            "Principal": "*",
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::<your-bucket-name>/*",
            "Condition": {
                "Null": {
                    "s3:x-amz-server-side-encryption-customer-algorithm": "false"
                }
            }
        },
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "s3:GetObjectVersion",
```

```
                "s3:ListBucketVersions",
                "s3:ListBucket",
                "s3:PutObjectLegalHold",
                "s3:GetBucketVersioning",
                "s3:GetObjectLegalHold",
                "s3:GetBucketObjectLockConfiguration",
                "s3:PutObject*",
                "s3:GetObject*",
                "s3:GetEncryptionConfiguration",
                "s3:PutObjectRetention",
                "s3:PutBucketObjectLockConfiguration",
                "s3:DeleteObject*",
                "s3:DeleteObjectVersion",
                "s3:GetBucketLocation"

            ],
            "Resource": [
                "arn:aws:s3:::<yourbucketname>",
                "arn:aws:s3:::<yourbucketname>/*"
            ]
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": [
                "s3:ListAllMyBuckets",
                "s3:ListBucket"
            ],
            "Resource": "arn:aws:s3:::*"
        }
    ]
}
```

Policy IAM per il livello di archiviazione

> **Note**
>
> Cambia il nome dei bucket S3 nella policy di esempio da `<yourbucketname>` al nome del bucket S3 che desideri utilizzare per i backup a livello di archiviazione Veeam.

Per utilizzare il VPC, la sottorete e i gruppi di sicurezza esistenti:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "S3Permissions",
            "Effect": "Allow",
            "Action": [
                "s3:DeleteObject",
                "s3:PutObject",
                "s3:GetObject",
                "s3:RestoreObject",
                "s3:ListBucket",
                "s3:AbortMultipartUpload",
                "s3:GetBucketVersioning",
                "s3:ListAllMyBuckets",
                "s3:GetBucketLocation",
                "s3:GetBucketObjectLockConfiguration",
                "s3:PutObjectRetention",
                "s3:GetObjectVersion",
                "s3:PutObjectLegalHold",
                "s3:GetObjectRetention",
                "s3:DeleteObjectVersion",
                "s3:ListBucketVersions"
            ],
            "Resource": [
                "arn:aws:s3:::<bucket-name>",
                "arn:aws:s3:::<bucket-name>/*"
            ]
        }
    ]
}

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "EC2Permissions",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:CreateKeyPair",
                "ec2:DescribeKeyPairs",
                "ec2:RunInstances",
```

```
                "ec2:DeleteKeyPair",
                "ec2:DescribeVpcAttribute",
                "ec2:CreateTags",
                "ec2:DescribeSubnets",
                "ec2:TerminateInstances",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeImages",
                "ec2:DescribeVpcs"
            ],
            "Resource": "arn:aws:ec2:<region>:<account-id>:*"
        }
    ]
}
```

Per creare nuovi VPC, sottorete e gruppi di sicurezza:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "S3Permissions",
            "Effect": "Allow",
            "Action": [
                "s3:DeleteObject",
                "s3:PutObject",
                "s3:GetObject",
                "s3:RestoreObject",
                "s3:ListBucket",
                "s3:AbortMultipartUpload",
                "s3:GetBucketVersioning",
                "s3:ListAllMyBuckets",
                "s3:GetBucketLocation",
                "s3:GetBucketObjectLockConfiguration",
                "s3:PutObjectRetention",
                "s3:GetObjectVersion",
                "s3:PutObjectLegalHold",
                "s3:GetObjectRetention",
                "s3:DeleteObjectVersion",
                "s3:ListBucketVersions"
            ],
            "Resource": [
                "arn:aws:s3:::<bucket-name>",
                "arn:aws:s3:::<bucket-name>/*"
```

```
            ]
        }
    ]
}

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "EC2Permissions",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:CreateKeyPair",
                "ec2:DescribeKeyPairs",
                "ec2:RunInstances",
                "ec2:DeleteKeyPair",
                "ec2:DescribeVpcAttribute",
                "ec2:CreateTags",
                "ec2:DescribeSubnets",
                "ec2:TerminateInstances",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeImages",
                "ec2:DescribeVpcs",
                "ec2:CreateVpc",
                "ec2:CreateSubnet",
                "ec2:DescribeAvailabilityZones",
                "ec2:CreateRoute",
                "ec2:CreateInternetGateway",
                "ec2:AttachInternetGateway",
                "ec2:ModifyVpcAttribute",
                "ec2:CreateSecurityGroup",
                "ec2:DeleteSecurityGroup",
                "ec2:AuthorizeSecurityGroupIngress",
                "ec2:AuthorizeSecurityGroupEgress",
                "ec2:DescribeRouteTables",
                "ec2:DescribeInstanceTypes"
            ],
            "Resource": "*"
        }
    ]
}
```

Copia i dati da un bucket Amazon S3 a un altro account e regione utilizzando AWS CLI

Appasaheb Bagali e Purushotham GK, Amazon Web Services

Riepilogo

Questo modello descrive come migrare i dati da un bucket Amazon Simple Storage Service (Amazon S3) di origine in un account a AWS un bucket Amazon S3 di destinazione in un altro AWS account, nella stessa regione o in un'altra regione. Regione AWS

Il bucket Amazon S3 di origine consente l'accesso AWS Identity and Access Management (IAM) utilizzando una policy di risorse allegata. Un utente nell'account di destinazione deve assumere un ruolo `PutObject` e le `GetObject` autorizzazioni per il bucket di origine. Infine, esegui `copy` i `sync` comandi per trasferire i dati dal bucket Amazon S3 di origine al bucket Amazon S3 di destinazione.

Gli account possiedono gli oggetti che caricano nei bucket Amazon S3. Se copi oggetti tra account e regioni, concedi all'account di destinazione la proprietà degli oggetti copiati. È possibile modificare la proprietà di un oggetto modificando la relativa lista di controllo d'accesso (ACL) in. `bucket-owner-full-control` Tuttavia, si consiglia di concedere autorizzazioni programmatiche per più account all'account di destinazione, poiché ACLs può essere difficile da gestire per più oggetti.

> **Warning**
>
> Questo scenario richiede agli utenti IAM un accesso programmatico e credenziali a lungo termine, il che rappresenta un rischio per la sicurezza. Per ridurre questo rischio, si consiglia di fornire a questi utenti solo le autorizzazioni necessarie per eseguire l'attività e di rimuoverli quando non sono più necessari. Le chiavi di accesso possono essere aggiornate se necessario. Per ulteriori informazioni, consulta Aggiornamento delle chiavi di accesso nella documentazione IAM.

Prerequisiti e limitazioni

Prerequisiti

- Due attivi Account AWS nello stesso modo o in modo diverso Regioni AWS.
- Un bucket Amazon S3 esistente nell'account di origine.

- Se il bucket Amazon S3 di origine o di destinazione ha la crittografia predefinita abilitata, devi modificare le autorizzazioni della chiave AWS Key Management Service (AWS KMS). Per ulteriori informazioni, consulta l'articolo di AWS re:Post su questo argomento.
- Familiarità con le autorizzazioni per più account.

Limitazioni

- Questo modello riguarda la migrazione una tantum. Per gli scenari che richiedono la migrazione continua e automatica di nuovi oggetti da un bucket di origine a un bucket di destinazione, puoi utilizzare Amazon S3 Batch Replication.
- Questo modello utilizza credenziali di sessione (`AccessKeyIdSecretAccessKey`, `eSessionToken`) temporanee e non persistenti. Il timestamp di scadenza nell'output indica quando scadono queste credenziali. Il ruolo è configurato con la durata massima della sessione. Il processo di copia verrà annullato se la sessione scade.

Architecture

Tools (Strumenti)

- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire Servizi AWS tramite comandi nella shell della riga di comando.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Best practice

- Le migliori pratiche di sicurezza in IAM (documentazione IAM)
- Applicazione delle autorizzazioni con privilegi minimi (documentazione IAM)

Epiche

Crea un utente e un ruolo IAM nella destinazione Account AWS

Operazione	Description	Competenze richieste
Crea un utente IAM e ottieni la chiave di accesso.	1. Accedi a Console di gestione AWS e crea un utente IAM con accesso programmatico. Per istruzioni dettagliate, consulta Creazione di utenti IAM nella documentazione IAM. Non è necessario allegare alcuna policy per questo utente. 2. Genera una chiave di accesso e una chiave segreta per questo utente. Per istruzioni, consulta Account AWS e accedi ai tasti nella AWS documenta zione.	AWS DevOps
Crea una policy basata sull'identità IAM.	Crea una policy basata sull'identità IAM denominat a `S3MigrationPolicy` utilizzando le seguenti autorizzazioni. Modifica i nomi dei bucket di origine e destinazione in base al tuo caso d'uso. Questa politica basata sull'identità consente all'utente che assume questo ruolo di accedere al bucket di origine e al bucket di destinazi one. Per istruzioni dettagliate,	AWS DevOps

Operazione	Description	Competenze richieste
	consulta Creazione di politiche IAM nella documentazione IAM. `{ "Version": "2012-10-17", "Statement": [{ "Effect": "Allow", "Action": ["s3:ListBucket", "s3:ListObjectsV2", "s3:GetObject", "s3:GetObjectTagging", "s3:GetObjectVersion", "s3:GetObjectVersionTagging"], "Resource": ["arn:aws:s3:::amazon-s3-demo-source-bucket", "arn:aws:s3:::amazon-s3-demo-source-bucket/*"] }, {`	

Operazione	Description	Competenze richieste
	"Effect": "Allow", "Action": ["s3:ListBucket", "s3:PutObject", "s3:PutObjectAcl", "s3:PutObjectTagging", "s3:GetObjectTagging", "s3:ListObjectsV2", "s3:GetObjectVersion", "s3:GetObjectVersionTagging"], "Resource": ["arn:aws:s3:::amazon-s3-demo-destination-bucket", "arn:aws:s3:::amazon-s3-demo-destination-bucket/*"] }] }	

Operazione	Description	Competenze richieste
Crea un ruolo IAM.	Crea un ruolo IAM denominato `S3MigrationRole` utilizzando la seguente politica di fiducia. Modifica l'Amazon Resource Name (ARN) del ruolo o del nome utente IAM di destinazione nella policy di fiducia in base al tuo caso d'uso. Questa politica di fiducia consente all'utente IAM appena creato di assumere`S3MigrationRole` . Allega il file creato in precedenza`S3MigrationPolicy` . Per i passaggi dettagliati, consulta Creazione di un ruolo per delegare le autorizzazioni a un utente IAM nella documentazione IAM. `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "AWS": "arn:aws:iam::<destination_account>:user/<user_name>" }, "Action": "sts:AssumeRole",`	AWS DevOps

Operazione	Description	Competenze richieste

	<pre>"Conditio
	n": {}
	}
	]
	}</pre>	

Crea e allega la policy del bucket Amazon S3 nell'account di origine

Operazione	Description	Competenze richieste
Crea e allega una policy sui bucket Amazon S3.	1. Accedi al tuo account Console di gestione AWS di origine e apri la console Amazon S3. 2. Scegli il bucket Amazon S3 di origine, quindi scegli Autorizzazioni. 3. In Policy del bucket, scegli Modifica. 4. Incolla la seguente policy sui bucket. Assicurati di includere l' Account AWS ID dell'account di destinazione e di configurare il modello di bucket policy in base alle tue esigenze. Questa policy basata sulle risorse consente al ruolo di destinazione di accedere `S3MigrationRole` agli oggetti Amazon S3 nell'account di origine. <pre>{</pre>	Amministratore cloud

Operazione	Description	Competenze richieste
	"Version": "2012-10-17", "Statement": [{ "Sid": "DelegateS3Access", "Effect": "Allow", "Principal": {"AWS": "arn:aws:iam::<destination_account>:role/<RoleName>"}, "Action": ["s3:ListBucket", "s3:GetObject", "s3:ListObjectsV2", "s3:GetObjectTagging", "s3:GetObjectVersion", "s3:GetObjectVersionTagging"], "Resource": ["arn:aws:s3:::amazon-s3-demo-source-bucket/*", "arn:aws:s3:::amazon-s3-demo-source-bucket"] }	

Operazione	Description	Competenze richieste
	] } 5. Scegli Save (Salva).	

Configurazione del bucket Amazon S3 di destinazione

Operazione	Description	Competenze richieste
Crea un bucket Amazon S3 di destinazione.	1. Accedi all'account Console di gestione AWS di destinazione e apri la console Amazon S3. 2. Seleziona Crea bucket. 3. Crea un bucket Amazon S3 in base alle tue esigenze. Per ulteriori informazioni, consulta Creazione di un bucket nella documenta zione di Amazon S3.	Amministratore cloud

Copia i dati nel bucket Amazon S3 di destinazione

Operazione	Description	Competenze richieste
Configuralo AWS CLI con le credenziali utente appena create.	1. Installa l'ultima versione di. AWS CLI Per istruzion i, vedere Installazione o aggiornamento della versione più recente di AWS CLI nella AWS CLI documentazione. 2. Esegui `$ aws configure` e aggiorna AWS CLI	AWS DevOps

Operazione	Description	Competenze richieste
	con la chiave di AWS accesso dell'utente IAM che hai creato. Per ulteriori informazioni, consulta Configurazione e impostazioni dei file di credenziali nella AWS CLI documentazione.	

Operazione	Description	Competenze richieste
Assumi il ruolo di migrazione di Amazon S3.	1. Usa il AWS CLI per `S3MigrationRole` supporre: `aws sts assume-role \ --role-arn "arn:aws:iam::<destination_account>:role/S3MigrationRole" \ --role-session-name AWSCLI-Session` **Note** Questo comando restituisce diverse informazioni. All'interno del blocco delle credenziali è necessario il comando `AccessKeyId SecretAccessKey` , e. `SessionToken` Questo esempio utilizza le variabili di ambiente `RoleAccessKeyID RoleSecretKey` , `eRoleSessionToken` .:	Amministratore AWS

Operazione	Description	Competenze richieste
	Le credenziali di sessione (`AccessKeyId SecretAccessKey , eSessionToken`) sono temporanee e non persistenti. Il timestamp di scadenza nell'output indica quando scadono queste credenziali. Il ruolo è configurato con la durata massima della sessione. Se le credenziali scadono, è necessario chiamare `sts:AssumeRole` nuovamente per ottenere nuove credenziali temporanee. 2. Crea tre variabili di ambiente per assumere il ruolo IAM. Queste variabili di ambiente vengono completate con il seguente output: `# Linux`	

Operazione	Description	Competenze richieste
	``` export AWS_ACCES S_KEY_ID=<RoleAcce ssKeyID from command output> export AWS_SECRE T_ACCESS_KEY=<Role SecretKey from command output> export AWS_SESSI ON_TOKEN=<RoleSess ionToken from command output> # Windows set AWS_ACCES S_KEY_ID=<RoleAcce ssKeyID from command output> set AWS_SECRE T_ACCESS_KEY=<Role SecretKey from command output> set AWS_SESSI ON_TOKEN=<RoleSess ionToken from command output> ``` 3. Verifica di aver assunto il ruolo IAM eseguendo il comando seguente: ``` aws sts get-caller- identity ``` Per ulteriori informazioni, consulta Come si usa il AWS CLI per assumere un ruolo IAM?	

Operazione	Description	Competenze richieste

Copia e sincronizza i dati dal bucket di origine al bucket di destinazione.	Una volta assunto il ruolo, `S3MigrationRole` è possibile copiare i dati utilizzando il comando copy (`cp`) o synchronize (). `sync`
	
	Copia:
	
	`aws s3 cp s3://amazon-s3-demo-source-bucket/ \ s3://amazon-s3-demo-destination-bucket/ \ --recursive --source-region SOURCE-REGION-NAME --region DESTINATION-REGION-NAME`
	
	Sincronizza:
	
	`aws s3 sync s3://amazon-s3-demo-source-bucket/ \ s3://amazon-s3-demo-destination-bucket/ \ --source-region SOURCE-REGION-NAME --region DESTINATION-REGION-NAME`
		Amministratore cloud

Risoluzione dei problemi

Problema	Soluzione
Si è verificato un errore (`AccessDenied`) durante la chiamata dell'`ListObjects` operazione	1. Assicurati di aver assunto il ruolo `S3MigrationRole`. 2. Esegui `aws sts get-caller-identity` per verificare il ruolo utilizzato. Se l'output non mostra l'ARN per `S3MigrationRole`, assumi nuovamente il ruolo IAM e riprova.

Risorse correlate

- Creazione di un bucket Amazon S3 (documentazione Amazon S3)
- Politiche dei bucket Amazon S3 e politiche degli utenti (documentazione Amazon S3)
- Identità IAM (utenti, gruppi e ruoli) (documentazione IAM)
- comando cp (AWS CLI documentazione)
- comando sync (AWS CLI documentazione)

Abilita l'archiviazione dei DB2 log direttamente su Amazon S3 in un database IBM Db2

Ambarish Satarkar, Amazon Web Services

Riepilogo

Questo modello descrive come utilizzare Amazon Simple Storage Service (Amazon S3) come storage del catalogo per i log di archivio generati da IBM Db2, senza utilizzare un'area di staging.

Puoi specificare lo storage DB2REMOTO di Amazon S3 per i parametri di configurazione dei metodi di archiviazione dei log logarchmeth1 e logarchmeth2. È possibile utilizzare il `logarchmeth1` parametro per specificare la destinazione principale per i log archiviati dal percorso di registro corrente. Con questa funzionalità, puoi archiviare e recuperare i log delle transazioni da e verso Amazon S3 direttamente, senza utilizzare un'area di staging.

Amazon S3 archivia i dati caricati su di esso su almeno tre dispositivi in un unico dispositivo. Regione AWS Milioni di clienti di tutte le dimensioni e settori utilizzano Amazon S3 per archiviare i backup aziendali grazie all'elevata disponibilità, alle opzioni di storage flessibili, alle politiche del ciclo di vita e alla sicurezza.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Database IBM Db2 in esecuzione su un'istanza Amazon Elastic Compute Cloud (Amazon EC2).
- AWS Command Line Interface () installato AWS CLI
- libcurl e libxml2 installati sull'istanza Db2. EC2

Limitazioni

- Solo Db2 11.5.7 o versione successiva consente l'archiviazione dei log direttamente nello storage Amazon S3.
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per informazioni sulla disponibilità per regione, vedi AWS Servizi per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.
- In tutte le configurazioni, esistono le seguenti limitazioni per Amazon S3:

- AWS Key Management Service (AWS KMS) non è supportato.
- AWS le credenziali basate su ruoli AWS Identity and Access Management ((IAM)) o basate su token (AWS Security Token Service (AWS STS)) non sono supportate.

Versioni del prodotto

- AWS CLI versione 2 o successiva
- IBM Db2 11.5.7 o versione successiva
- Linux SUSE Linux Enterprise Server (SLES) 11 o versione successiva
- Red Hat Enterprise Linux (RHEL) 6 o versione successiva
- Windows Server 2008 R2, 2012 (R2), 2016 o 2019

Architecture

Il diagramma seguente mostra i componenti e il flusso di lavoro di questo modello.

L'architettura del Cloud AWS include quanto segue:

- Cloud privato virtuale (VPC): una sezione logicamente isolata in Cloud AWS cui vengono lanciate le risorse.
- Zona di disponibilità: fornisce un'elevata disponibilità eseguendo il carico di lavoro Db2 LUW (Linux, Unix, Windows) in un data center isolato all'interno di. Regione AWS
- Sottorete pubblica: fornisce l'accesso RDP (Remote Desktop Protocol) per gli amministratori e la connettività Internet tramite un gateway NAT.
- Sottorete privata: ospita il database Db2 LUW. L'istanza Db2 LUW è configurata con il parametro. `LOGARCHMETH1` Il parametro scrive i file di archivio dei log del database direttamente su un percorso Amazon S3 attraverso l'endpoint del gateway.

Quanto segue Servizi AWS fornisce supporto:

- Amazon S3: funge da posizione di archiviazione durevole e scalabile per i file di archivio di log Db2.
- Amazon Elastic File System (Amazon EFS): fornisce un file system condiviso e completamente gestito che Db2 può utilizzare per i backup e lo staging del database. Db2 può anche utilizzare Amazon EFS come punto di montaggio per i file di log prima che vengano archiviati in Amazon S3.

- Amazon CloudWatch: raccoglie e monitora metriche, log ed eventi da Db2 e dalle istanze sottostanti. EC2 Puoi utilizzarlo CloudWatch per creare allarmi, dashboard e risposte automatiche a problemi di prestazioni o disponibilità.

Automazione e scalabilità

- Questo modello fornisce una soluzione completamente automatizzata per archiviare il backup degli archivi di log Db2.
- Puoi utilizzare lo stesso bucket Amazon S3 per abilitare l'archiviazione dei log di più database Db2.

Tools (Strumenti)

Servizi AWS

- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue AWS risorse e delle applicazioni su cui esegui AWS in tempo reale.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire Servizi AWS tramite comandi nella shell della riga di comando.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon Elastic File System (Amazon EFS) ti aiuta a creare e configurare file system condivisi nel Cloud AWS.
- AWS IAM Identity Centerti aiuta a gestire centralmente l'accesso Single Sign-On (SSO) a tutte le tue applicazioni e al cloud. Account AWS
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.

Altri strumenti

- libcurl è una libreria gratuita per il trasferimento di URL lato client.
- libxml2 è un parser e un toolkit XML C gratuiti.

Best practice

- Segui il principio del privilegio minimo e concedi le autorizzazioni minime richieste per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.

Poemi epici

Configurazione Servizi AWS

Operazione	Description	Competenze richieste

Configura il AWS CLI.	Per scaricare e installare AWS CLI, utilizzare i seguenti comandi:
	`i) curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"`
	`ii) unzip awscliv2.zip`
	`iii) sudo ./aws/install`
		Amministratore di sistema AWS, amministratore AWS

Configurare AWS CLI.	Per configurare AWS CLI, usa i seguenti comandi:
	`$ aws configure`
	`AWS Access Key ID [None]:**********`
	`**********`
	`AWS Secret Access Key [None]: *********`
	`**********`
	`Default region name [None]: < aws region >`
	`Default output format [None]: text`
		Amministratore di sistema AWS, amministratore AWS

Operazione	Description	Competenze richieste

Crea utente IAM.	Per creare un utente IAM da utilizzare in seguito per la connessione al database Db2 con Amazon S3, usa il seguente comando:
	
	`aws iam create-user --user-name <unique username>`
	
	Di seguito è riportato un esempio del comando:
	
	`aws iam create-user --user-name db_backup_user`
	
	Warning
	Questo scenario richiede agli utenti IAM un accesso programmatico e credenziali a lungo termine, il che rappresenta un rischio per la sicurezza. Per mitigare questo rischio, si consiglia di fornire a questi utenti solo le autorizzazioni necessarie per eseguire l'operazione e di rimuoverli quando non sono più necessari. Le chiavi
		Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	di accesso possono essere aggiornate se necessario. Per ulteriori informazioni, consulta le credenziali AWS di sicurezza e la gestione delle chiavi di accesso per gli utenti IAM nella documentazione IAM.	

Crea un bucket Amazon S3.	Per creare un bucket Amazon S3 per l'archiviazione del backup del database, usa il seguente comando:
	`aws s3api create-bucket --bucket <unique bucket name> --region <aws region>`
	Di seguito è riportato un comando di esempio.
	`aws s3api create-bucket --bucket myfirstbucket --region af-south-1`
		Amministratore di sistema AWS

Operazione	Description	Competenze richieste
Autorizza l'utente IAM.	Per autorizzare l'utente IAM appena creato a disporre delle autorizzazioni Amazon S3, utilizza i seguenti passaggi: 1. Per generare un nuovo file di policy IAM `db2bucket.json` per il bucket Amazon S3, usa il seguente comando: `{ "Version": "2012-10-17", "Statement": [ { "Sid": "ListObjectsInBucket", "Effect": "Allow", "Action": "s3:ListBucket", "Resource": [ "arn:aws:s3:::myfirstbucket" ] }, { "Sid": "AllObjectActions", "Effect": "Allow", "Action": [ "s3:PutObject",`	Amministratore di sistema AWS, amministratore AWS

Operazione	Description	Competenze richieste

```
 "s3:GetObject",

 "s3:DeleteObject"
           ],
           "Resource
": [

 "arn:aws:s3:::myfi
rstbucket/*"
           ]
       }
    ]
}
```

2. Crea una policy IAM basata sul `db2bucket.json` file. L'output JSON mostra l'Amazon Resource Name (ARN) per la policy:. `"Arn": "arn:aws:iam::111122223333:policy/db2s3policy"`

```
aws iam create-policy
 \
      --policy-name
 db2s3policy \
      --policy-
document file://db
2bucket.json
```

3. Collega la policy IAM all'utente IAM con la policy ARN.

```
aws iam attach-user-
policy --user-name
```

Operazione	Description	Competenze richieste
	db_backup_user -- policy-arn "arn:aws: iam::111122223333: policy/db2s3policy "	

Operazione	Description	Competenze richieste

Crea una chiave di accesso.	Per generare una chiave di accesso per accedere programmaticamente ad Amazon S3 dall'istanza, usa DB2 il seguente comando:
	
	`aws iam create-access-key --user-name <username>`
	
	Di seguito è riportato un esempio del comando:
	
	`aws iam create-access-key --user-name db_backup_user`
	
	Warning
	Questo scenario richiede agli utenti IAM un accesso programmatico e credenziali a lungo termine, il che rappresenta un rischio per la sicurezza. Per mitigare questo rischio, si consiglia di fornire a questi utenti solo le autorizzazioni necessarie per eseguire l'operazione e di rimuoverli quando non sono più necessari. Le chiavi
		Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	di accesso possono essere aggiornate se necessario. Per ulteriori informazioni, consulta le credenziali AWS di sicurezza e la gestione delle chiavi di accesso per gli utenti IAM nella documentazione IAM.	
Crea un keystore PKCS.	Per creare un keystore PKCS per archiviare la chiave e creare una chiave di accesso segreta per trasferire i dati su Amazon S3, usa il seguente comando: `gsk8capicmd_64 -keydb -create -db "/db2/db2<sid>/.keystore/db6-s3.p12" -pw "<password>" -type pkcs12 -stash`	Amministratore di sistema AWS

Operazione	Description	Competenze richieste
Configura DB2 per utilizzare il keystore.	DB2 Per configurare l'utilizzo del keystore con i `keystore_type` parametri `keystore_location` and, utilizzate i seguenti comandi: `db2 "update dbm cfg using keystore_location /db2/db2<sid>/.keystore/db6-s3.p12 keystore_type pkcs12"`	Amministratore di sistema AWS

Operazione	Description	Competenze richieste

Crea un alias DB2 di accesso allo storage.	Un alias di accesso allo storage specifica il bucket Amazon S3 da utilizzare. Fornisce inoltre i dettagli di connessione, come nome utente e password, archiviati nel keystore locale in un formato crittografato. Per ulteriori informazioni, consulta il comando CATALOG STORAGE ACCESS nella documentazione IBM Db2.
	
	Per creare un alias di accesso allo storage, utilizza la seguente sintassi:
	
	`db2 "catalog storage access alias <alias_name> vendor S3 server <S3 endpoint> user '<access_key>' password '<secret_access_key>' container '<bucket_name>'"`
	
	Di seguito è riportato un esempio:
	
	`db2 "catalog storage access alias DB2BKPS3 vendor S3 server s3.us-west-2.amazonaws.com user '********** ********** ' password '**********`
		Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	`****' container 'myfirstbucket'"`	

Aggiorna la posizione logarchmeth1 e riavvia DB2 DB2

Operazione	Description	Competenze richieste

Aggiorna la `LOGARCHMETH1` posizione.	Per utilizzare l'alias di accesso allo storage definito in precedenza, aggiornare i parametri del `LOGARCHMETH1` database, utilizzare il seguente comando:
	`db2 update db cfg for <DBNAME> using LOGARCHMETH1 'DB2REMOTE://<storage_alias_name>//<sub folder>'`
	Per separare i log dagli altri file, specifica una sottodirectory (ovvero il prefisso del bucket Amazon `S3TESTDB_LOGS`) in cui salvare i log all'interno del bucket S3.
	Di seguito è riportato un esempio:
	`db2 update db cfg for ABC using LOGARCHMETH1 'DB2REMOTE://DB2BKPS3//TESTDB_LOGS/'`
		Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	Dovresti vedere il seguente messaggio: `DB20000I The UPDATE DATABASE CONFIGURATION command completed successfully.`	
Riavvia DB2.	Riavvia l' DB2 istanza dopo averla riconfigurata per l'archiviazione dei log. Tuttavia, se in precedenza `LOGARCHMETH1` era impostato su una posizione del file system, il riavvio non è necessario.	Amministratore AWS, amministratore di sistema AWS

Controlla il percorso del log di archivio in Amazon S3 e db2diag.log

Operazione	Description	Competenze richieste
Controlla il log di archivio in Amazon S3.	A questo punto, il database è completamente configurato per archiviare i log delle transazioni direttamente nello storage Amazon S3. Per confermare la configurazione, inizia a eseguire attività transazionali sul database per iniziare a consumare (e archiviare) lo spazio dei log. Quindi, controlla i log di archivio in Amazon S3.	Amministratore AWS, amministratore di sistema AWS

Operazione	Description	Competenze richieste
Controlla la configurazione dei log di archivio indb2diag.log .	Dopo aver controllato il registro di archivio in Amazon S3, cerca il seguente messaggio nel registro di DB2 diagnostica: db2diag.log MESSAGE : ADM1846I Completed archive for log file "S0000079.LOG" to Comple ted archive for log file S0000080.LOG to DB2REMOTE://<AWS S3 Bucket Name>/<SI D>/log1/db2<sid>/< SID>/NODE0000/LOGS TREAM0000/C0000001 / from /db2/<SID>/ log_dir/NODE0000/ LOGSTREAM0000/. M ESSAGE : ADM1846I Completed archive for log file "S0000080.LOG" to Comple ted archive for log file S0000081.LOG to DB2REMOTE://<AWS S3 Bucket Name> / <SID>/log1/db2<si d>/<SID>/NODE0000/ LOGSTREAM0000/C000 0001/ from /db2/<SID >/log_dir/NODE0000/ LOGSTREAM0000/.	Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	Questo messaggio conferma che i file di log DB2 delle transazioni chiuse vengono archiviati nello storage Amazon S3 (remoto).	

Risorse correlate

AWS documentazione del servizio

- AWS credenziali di sicurezza (documentazione IAM)
- Concedi il privilegio minimo (documentazione IAM)
- Gestisci le chiavi di accesso per gli utenti IAM (documentazione IAM)
- Le migliori pratiche di sicurezza in IAM (documentazione IAM)

Risorse IBM

- Database IBM Db2
- logarchmeth1 - Parametro di configurazione del metodo di archiviazione dei log primario
- logarchmeth2 - Parametro di configurazione del metodo di archiviazione dei log secondario
- Archiviazione remota

Esegui la migrazione dei dati da un ambiente Hadoop locale ad Amazon S3 utilizzando AWS per Amazon S3 DistCp PrivateLink

Jason Owens, Andres Cantor, Jeff Klopfenstein, Bruno Rocha Oliveira e Samuel Schmidt, Amazon Web Services

Riepilogo

Questo modello dimostra come migrare quasi qualsiasi quantità di dati da un ambiente Apache Hadoop locale al cloud Amazon Web Services (AWS) utilizzando lo strumento open source Apache con DistCpAWS PrivateLink per Amazon Simple Storage Service (Amazon S3). Invece di utilizzare la rete Internet pubblica o una soluzione proxy per migrare i dati, puoi utilizzare AWS PrivateLink per Amazon S3 per migrare i dati su Amazon S3 tramite una connessione di rete privata tra il tuo data center locale e un Amazon Virtual Private Cloud (Amazon VPC). Se utilizzi voci DNS in Amazon Route 53 o aggiungi voci nel file /etc/hosts in tutti i nodi del tuo cluster Hadoop locale, verrai indirizzato automaticamente all'endpoint di interfaccia corretto.

Questa guida fornisce istruzioni per l'uso DistCp per la migrazione dei dati nel cloud AWS. DistCp è lo strumento più comunemente usato, ma sono disponibili altri strumenti di migrazione. Ad esempio, puoi utilizzare strumenti AWS offline come AWS Snowball o AWS Snowmobileo strumenti AWS online come AWS Storage Gateway o AWS. DataSync Inoltre, puoi utilizzare altri strumenti open source come Apache. NiFi

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo con una connessione di rete privata tra il data center locale e il cloud AWS
- Hadoop, installato in locale con DistCp
- Un utente Hadoop con accesso ai dati di migrazione nell'Hadoop Distributed File System (HDFS)
- AWS Command Line Interface (AWS CLI), installata e configurata
- Autorizzazioni per inserire oggetti in un bucket S3

Limitazioni

Le limitazioni del cloud privato virtuale (VPC) si applicano ad AWS PrivateLink per Amazon S3. Per ulteriori informazioni, consulta Proprietà e limitazioni degli endpoint dell'interfaccia e PrivateLink quote AWS (PrivateLink documentazione AWS).

AWS PrivateLink per Amazon S3 non supporta quanto segue:

- Endpoint FIPS (Federal Information Processing Standard)
- Endpoint del sito Web
- Endpoint globali legacy

Architecture

Stack tecnologico di origine

- Cluster Hadoop con installato DistCp

Stack tecnologico Target

- Simple Storage Service (Amazon S3)
- Amazon VPC

Architettura di destinazione

Il diagramma mostra come l'amministratore Hadoop utilizza DistCp per copiare i dati da un ambiente locale tramite una connessione di rete privata, come AWS Direct Connect, ad Amazon S3 tramite un endpoint di interfaccia Amazon S3.

Tools (Strumenti)

Servizi AWS

- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito. Questa rete virtuale è simile a una rete tradizionale che gestiresti nel tuo data center, con i vantaggi dell'utilizzo dell'infrastruttura scalabile di AWS.

Altri strumenti

- Apache Hadoop DistCp (copia distribuita) è uno strumento utilizzato per copiare intercluster e intracluster di grandi dimensioni. DistCp utilizza MapReduce Apache per la distribuzione, la gestione e il ripristino degli errori e la segnalazione.

Epiche

Migrazione dei dati nel cloud AWS

Operazione	Description	Competenze richieste
Crea un endpoint per AWS PrivateLink per Amazon S3.	1. Accedi alla Console di gestione AWS e apri la console Amazon VPC. 2. Nel pannello di navigazione, scegli Endpoints, quindi scegli Crea endpoint. 3. Per Service category (Categoria servizio), scegli AWS services (Servizi AWS). 4. Nella casella di ricerca, inserisci s3, quindi premi Invio. 5. Nei risultati della ricerca, scegli com.amazonaws.< your-aws-region >.s3 nome di servizio in cui il valore nella colonna Tipo è Interfaccia. 6. In VPC, seleziona il VPC. Per Sottoreti, scegli le tue sottoreti. 7. Per il gruppo di sicurezza, scegli o crea un gruppo di	Amministratore AWS

Operazione	Description	Competenze richieste
	sicurezza che consenta il protocollo TCP 443. 8. Aggiungi tag in base alle tue esigenze, quindi scegli Crea endpoint.	
Verifica gli endpoint e trova le voci DNS.	1. Apri la console Amazon VPC, scegli Endpoints, quindi seleziona l'endpoint che hai creato in precedenza. 2. Nella scheda Dettagli, trova la prima voce DNS per i nomi DNS. Questa è la voce DNS regionale. Quando si utilizza questo nome DNS, le richieste si alternano tra le voci DNS specifiche delle zone di disponibilità. 3. Scegli la scheda Subnet. È possibile trovare l'indirizzo dell'interfaccia di rete elastica dell'endpoint in ciascuna zona di disponibilità.	Amministratore AWS

Operazione	Description	Competenze richieste

Controlla le regole del firewall e le configurazioni di routing.	Per verificare che le regole del firewall siano aperte e che la configurazione di rete sia impostata correttamente, utilizzate Telnet per testare l'endpoint sulla porta 443. Esempio:
	`$ telnet vpce-<your-VPC-endpoint-ID>.s3.us-east-2.vpce.amazonaws.com 443`
	`Trying 10.104.88.6...`
	`Connected to vpce-<your-VPC-endpoint-ID>.s3.us-east-2.vpce.amazonaws.com.`
	`...`
	`$ telnet vpce-<your-VPC-endpoint-ID>.s3.us-east-2.vpce.amazonaws.com 443`
	`Trying 10.104.71.141...`
	`Connected to vpce-<your-VPC-endpoint-ID>.s3.us-east-2.vpce.amazonaws.com.`
	Note
	Se utilizzi la voce Regionale, un test
		Amministratore di rete, amministratore AWS

Operazione	Description	Competenze richieste
	riuscito mostra che il DNS si alterna tra i due indirizzi IP che puoi vedere nella scheda Subnet per l'endpoint selezionato nella console Amazon VPC.	

Operazione	Description	Competenze richieste
Configura la risoluzione dei nomi.	È necessario configura re la risoluzione dei nomi per consentire a Hadoop di accedere all'endpoint dell'interfaccia Amazon S3. Non è possibile utilizzare il nome dell'endpoint stesso. È invece necessario risolvere `<your-bucket-name> .s3.<your-aws-regi on>.amazonaws.com o*.s3.<your-aws-reg ion>.amazonaws.com` . Per ulteriori informazi oni su questa limitazione di denominazione, vedere Introduzione al client Hadoop S3A (sito Web Hadoop). Scegliete una delle seguenti opzioni di configurazione: • Utilizza il DNS locale per risolvere l'indirizzo IP privato dell'endpoint. È possibile sovrascrivere il comportamento di tutti i bucket o di quelli seleziona ti. Per ulteriori informazi oni, consulta «Opzione 2: accesso ad Amazon S3 utilizzando Domain Name System Response Policy Zones (DNS RPZ)» in Accesso ibrido sicuro ad	Amministratore AWS

Operazione	Description	Competenze richieste
	Amazon S3 usando AWS (PrivateLinkpost sul blog AWS). • Configura il DNS locale per inoltrare in modo condizionale il traffico agli endpoint in entrata del resolver nel VPC. Il traffico viene inoltrato alla Route 53. Per ulteriori informazioni, consulta «Opzione 3: inoltro di richieste DNS da locale utilizzando Amazon Route 53 Resolver Inbound Endpoints» in Accesso ibrido sicuro ad Amazon S3 usando AWS (post sul blog AWS). PrivateLink • Modifica il file /etc/hosts su tutti i nodi del tuo cluster Hadoop. Questa è una soluzione temporanea per i test e non è consigliata per la produzione. Per modificare il file /etc/hosts, aggiungete una voce per una delle due opzioni o. `<your-bucket-name>.s3.<your-aws-region>.amazonaws.com s3.<your-aws-region>.amazonaws.com` Il file /etc/hosts non può avere più indirizzi IP per una voce.	

Operazione	Description	Competenze richieste
	È necessario scegliere un singolo indirizzo IP da una delle zone di disponibilità, che diventa quindi un singolo punto di errore.	

Operazione	Description	Competenze richieste
Configura l'autenticazione per Amazon S3.	Per l'autenticazione su Amazon S3 tramite Hadoop, consigliamo di esportare le credenziali temporanee dei ruoli nell'ambiente Hadoop. Per ulteriori informazioni, consulta Autenticazione con S3 (sito Web Hadoop). Per i lavori di lunga durata, puoi creare un utente e assegnare una policy con le autorizza zioni per inserire i dati solo in un bucket S3. La chiave di accesso e la chiave segreta possono essere archiviate su Hadoop, accessibili solo al DistCp lavoro stesso e all'amministratore Hadoop. Per ulteriori informazioni sull'arch iviazione dei segreti, vedere Archiviazione dei segreti con i provider di credenziali Hadoop (sito Web Hadoop). Per ulteriori informazioni su altri metodi di autenticazione, consulta How to get credentia ls of an IAM role for use with CLI access a un account AWS nella documentazione di AWS IAM Identity Center (successo re di AWS Single Sign-On). Per utilizzare credenzia li temporanee, aggiungi le credenziali temporane	Amministratore AWS

Operazione	Description	Competenze richieste

	e al file delle credenziali o esegui i seguenti comandi per esportare le credenziali nel tuo ambiente:
	`export AWS_SESSION_TOKEN=SECRET-SESSION-TOKEN`
	`export AWS_ACCESS_KEY_ID=SESSION-ACCESS-KEY`
	`export AWS_SECRET_ACCESS_KEY=SESSION-SECRET-KEY`
	Se disponi di una combinazione di chiave di accesso tradizionale e chiave segreta, esegui i seguenti comandi:
	`export AWS_ACCESS_KEY_ID=my.aws.key`
	`export AWS_SECRET_ACCESS_KEY=my.secret.key`
	Note
	Se utilizzi una combinazione di chiave di accesso e chiave segreta, modifica il fornitore delle credenziali nei DistCp comandi da `"org.apache.hadoop`	

Operazione	Description	Competenze richieste
	`.fs.s3a.TemporaryAWSCredentialsProvider" a"org.apache.hadoop.fs.s3a.SimpleAWSCredentialsProvider" .`	

Operazione	Description	Competenze richieste
Trasferisci dati utilizzando DistCp.	Da utilizzare DistCp per trasferire dati, esegui i seguenti comandi: `hadoop distcp -Dfs.s3a.aws.credentials.provider=\"org.apache.hadoop.fs.s3a.TemporaryAWSCredentialsProvider" \ -Dfs.s3a.access.key="${AWS_ACCESS_KEY_ID}" \ -Dfs.s3a.secret.key="${AWS_SECRET_ACCESS_KEY}" \ -Dfs.s3a.session.token="${AWS_SESSION_TOKEN}" \ -Dfs.s3a.path.style.access=true \ -Dfs.s3a.connection.ssl.enabled=true \ -Dfs.s3a.endpoint=s3.<your-aws-region>.amazonaws.com \ hdfs:///user/root/ s3a://<your-bucket-name>` **Note** La regione AWS dell'endpoint non viene rilevata automaticamente quando usi il DistCp comando con	Ingegnere addetto alla migrazione, amministratore AWS

Operazione	Description	Competenze richieste
	AWS PrivateLink per Amazon S3. Hadoop 3.3.2 e versioni successive risolvono questo problema abilitando l'opzione per impostare in modo esplicito la regione AWS del bucket S3. Per ulteriori informazioni, consulta S3A per aggiungere l'opzione fs.s3a.endpoint.region per impostare la regione AWS (sito Web Hadoop). Per ulteriori informazioni su provider S3A aggiuntivi, consulta Configurazione generale del client S3A (sito Web Hadoop). Ad esempio, se utilizzi la crittografia, puoi aggiungere la seguente opzione alla serie di comandi di cui sopra a seconda del tipo di crittografia: `-Dfs.s3a.server-side-encryption-algorithm=AES-256 [or SSE-C or SSE-KMS]`	

Operazione	Description	Competenze richieste
	Note Per utilizzare l'endpoint di interfaccia con S3A, è necessario creare una voce alias DNS per il nome regionale S3 (ad esempio, `s3.<your-aws-region>.amazonaws.com`) sull'endpoint dell'interfaccia. Per istruzioni, consulta la sezione Configurazione dell'autenticazione per Amazon S3. Questa soluzione alternativa è necessaria per Hadoop 3.3.2 e versioni precedenti. Le versioni future di S3A non richiederanno questa soluzione alternativa. Se hai problemi di firma con Amazon S3, aggiungi un'opzione per utilizzare la firma Signature Version 4 (SigV4): `-Dmapreduce.map.java.opts="-Dcom.ama`	

Operazione	Description	Competenze richieste
	`zonaws.services.s3 .enableV4=true"`	

Altri modelli

- Accesso Servizi AWS da IBM z/OS installando AWS CLI
- Archivia automaticamente gli elementi su Amazon S3 utilizzando DynamoDB TTL
- Esegui automaticamente il backup dei database SAP HANA utilizzando Systems Manager e EventBridge
- Esegui il backup e l'archiviazione dei dati del mainframe su Amazon S3 utilizzando BMC AMI Cloud Data
- Crea una pipeline di servizi ETL per caricare i dati in modo incrementale da Amazon S3 ad Amazon Redshift utilizzando AWS Glue
- Configura la registrazione delle chiamate dei modelli in Amazon Bedrock utilizzando AWS CloudFormation
- Converti e decomprimi i dati EBCDIC in ASCII su AWS usando Python
- Converti VARCHAR2 (1) il tipo di dati per Oracle in tipo di dati booleano per Amazon Aurora PostgreSQL
- Crea una definizione di attività Amazon ECS e monta un file system su EC2 istanze utilizzando Amazon EFS
- Distribuisci i record DynamoDB ad Amazon S3 utilizzando Kinesis Data Streams e Firehose con AWS CDK
- Implementa un file system Lustre per l'elaborazione dei dati ad alte prestazioni utilizzando Terraform e DRA
- Stima dei costi di storage per una tabella Amazon DynamoDB
- Identifica i bucket Amazon S3 pubblici utilizzando Security Hub (AWS Organizations CSPM)
- Esegui la migrazione di un server SFTP locale a Using AWS AWS Transfer for SFTP
- Esegui la migrazione di una tabella partizionata Oracle su PostgreSQL utilizzando AWS DMS
- Esegui la migrazione dei dati da Microsoft Azure Blob ad Amazon S3 utilizzando Rclone
- Esegui la migrazione dei valori Oracle CLOB su singole righe in PostgreSQL su AWS
- Migrazione di file system condivisi in una migrazione di AWS grandi dimensioni
- Esegui la migrazione di piccoli set di dati da locale ad Amazon S3 utilizzando AWS SFTP
- Monitora Amazon Aurora per le istanze senza crittografia
- Sposta i file mainframe direttamente su Amazon S3 utilizzando Transfer Family

- Rimuovi EC2 le voci Amazon nello stesso Account AWS modulo AWS Managed Microsoft AD utilizzando l' AWS Lambda automazione
- Esegui carichi di lavoro con stato con storage persistente dei dati utilizzando Amazon EFS su Amazon EKS con AWS Fargate
- Importa con successo un bucket S3 come stack AWS CloudFormation
- Sincronizza i dati tra i file system Amazon EFS in diverse regioni AWS utilizzando AWS DataSync
- Visualizza i dettagli degli snapshot EBS per il tuo account o la tua organizzazione AWS

Strumenti per sviluppatori

Argomenti

- DevOps
- Infrastruttura
- App Web e mobili

DevOps

Argomenti

- Accelera MLOps con Backstage e modelli Amazon SageMaker AI self-service
- Automatizza le operazioni AWS dell'infrastruttura utilizzando Amazon Bedrock
- Automatizza CloudFront gli aggiornamenti quando gli endpoint del bilanciamento del carico cambiano utilizzando Terraform
- Automatizza le CodeGuru revisioni Amazon per le applicazioni AWS CDK Python utilizzando Actions GitHub
- Automatizza l'implementazione dei Catena di approvvigionamento di AWS data lake in una configurazione multi-repository utilizzando GitHub Actions, Artifactory e Terraform
- Automatizza la valutazione delle risorse AWS
- Installa automaticamente i sistemi SAP utilizzando strumenti open source
- Automatizza il portafoglio e la distribuzione dei prodotti di AWS Service Catalog utilizzando AWS CDK
- Automatizza la gestione dinamica delle pipeline per l'implementazione di soluzioni hotfix in ambienti Gitflow utilizzando e AWS Service Catalog AWS CodePipeline
- Eliminazione automatica degli AWS CloudFormation stack e delle risorse associate
- Automatizza l'inserimento e la visualizzazione di metriche personalizzate di Amazon MWAA su Amazon Managed Grafana utilizzando Terraform
- Automatizza la distribuzione di stack set utilizzando AWS e AWS CodePipeline CodeBuild
- Associa automaticamente una policy gestita da AWS per Systems Manager ai profili di EC2 istanza utilizzando Cloud Custodian e AWS CDK
- Crea automaticamente CI/CD pipeline e cluster Amazon ECS per microservizi utilizzando AWS CDK
- Crea e invia immagini Docker ad Amazon ECR utilizzando GitHub Actions e Terraform
- Crea e testa app iOS con AWS CodeCommit, AWS e CodePipeline AWS Device Farm
- Configura l'autenticazione TLS reciproca per le applicazioni in esecuzione su Amazon EKS
- Automatizza la creazione di risorse Amazon WorkSpaces Applications utilizzando AWS CloudFormation
- Crea un parser di log personalizzato per Amazon ECS utilizzando un router di log Firelens

- Crea un framework di orchestrazione delle risorse basato su API utilizzando Actions e Terragrunt GitHub
- Crea richieste pull automatizzate per l' AWS infrastruttura gestita da Terraform utilizzando Actions GitHub
- Crea automaticamente pipeline CI dinamiche per progetti Java e Python
- Implementa i canarini CloudWatch Synthetics utilizzando Terraform
- Implementa una ChatOps soluzione per gestire i risultati della scansione SAST utilizzando Amazon Q Developer nelle applicazioni di chat, azioni personalizzate e CloudFormation
- Implementa sistemi agentici su Amazon Bedrock con il framework CrewAI utilizzando Terraform
- Implementa un job AWS Glue con una pipeline CodePipeline CI/CD AWS
- Distribuisci codice in più regioni AWS utilizzando AWS CodePipeline CodeCommit, AWS e AWS CodeBuild
- Distribuisci carichi di lavoro dalle DevOps pipeline di Azure ai cluster Amazon EKS privati
- Esegui le query SQL di Amazon Redshift utilizzando Terraform
- Esporta i report di AWS Backup da tutta l'organizzazione in AWS Organizations come file CSV
- Esporta i tag per un elenco di EC2 istanze Amazon in un file CSV
- Genera un CloudFormation modello AWS contenente le regole gestite di AWS Config utilizzando Troposphere
- Offri alle istanze di SageMaker notebook l'accesso temporaneo a un CodeCommit repository in un altro account AWS
- Implementa una strategia di ramificazione GitHub Flow per ambienti con più account DevOps
- Implementa una strategia di ramificazione Gitflow per ambienti con più account DevOps
- Implementa una strategia di ramificazione Trunk per ambienti con più account DevOps
- Implementa la scansione Checkov centralizzata e personalizzata per applicare le policy prima di implementare l'infrastruttura AWS
- Implementa la diagnostica e la risoluzione dei problemi di Kubernetes basate sull'intelligenza artificiale con K8SGPT e l'integrazione con Amazon Bedrock
- Rileva automaticamente le modifiche e avvia diverse CodePipeline pipeline per un monorepo in CodeCommit
- Integra un repository Bitbucket con AWS Amplify utilizzando AWS CloudFormation
- Avvia un CodeBuild progetto su più account AWS utilizzando Step Functions e una funzione proxy Lambda

- Gestione del failover Multi-AZ per i cluster EMR utilizzando Application Recovery Controller
- Gestisci le blue/green distribuzioni di microservizi su più account e regioni utilizzando i servizi di codice AWS e le chiavi multiregionali AWS KMS
- Monitora i repository Amazon ECR per le autorizzazioni wildcard utilizzando AWS e AWS Config CloudFormation
- Ottimizza le implementazioni serverless multi-account utilizzando i flussi di lavoro e Actions AWS CDK GitHub
- Esegui il provisioning di AWS Service Catalog prodotti basati su AWS CloudFormation modelli utilizzando GitHub Actions
- Fornisci ruoli IAM con privilegi minimi implementando una soluzione di distribuzione automatica di ruoli
- Pubblica i CloudWatch parametri di Amazon in un file CSV
- Rimuovi Amazon EC2 Entries Account AWS dall'altra AWS Managed Microsoft AD parte utilizzando l' AWS Lambda automazione
- Rimuovi EC2 le voci Amazon nello stesso Account AWS modulo AWS Managed Microsoft AD utilizzando l' AWS Lambda automazione
- Esegui test unitari per i lavori ETL in Python AWS Glue utilizzando il framework pytest
- Configura una CI/CD pipeline utilizzando AWS CodePipeline e AWS CDK
- Configura la registrazione centralizzata su scala aziendale utilizzando Terraform
- Configura end-to-end la crittografia per le applicazioni su Amazon EKS utilizzando cert-manager e Let's Encrypt
- Semplifica la distribuzione di applicazioni multi-tenant Amazon EKS utilizzando Flux
- Semplifica lo sviluppo e la distribuzione di bot Amazon Lex utilizzando un flusso di lavoro automatizzato
- Coordina la dipendenza dalle risorse e l'esecuzione delle attività utilizzando il costrutto AWS Fargate WaitCondition hook
- Usa repository di sorgenti Git di terze parti in AWS CodePipeline
- Crea una CI/CD pipeline per convalidare le configurazioni Terraform utilizzando AWS CodePipeline
- Altri modelli

Accelera MLOps con Backstage e modelli Amazon SageMaker AI self-service

Ashish Bhatt, Shashank Hiremath e Shivanshu Suryakar, Amazon Web Services

Riepilogo

Organizations che utilizzano sistemi di machine learning operations (MLOps) affrontano sfide significative in termini di scalabilità, standardizzazione e protezione della propria infrastruttura ML. Questo modello introduce un approccio trasformativo che combina Backstage, un portale per sviluppatori open source, con Amazon SageMaker AI e moduli Infrastructure as Code (IaC) rafforzati per migliorare il modo in cui i team di data science possono sviluppare, distribuire e gestire i flussi di lavoro ML.

I moduli IAc per questo pattern sono disponibili nell'archivio dei moduli. GitHub AWS AIOps Questi moduli offrono modelli predefiniti per configurare l'infrastruttura ML e creare ambienti ML coerenti. Tuttavia, i data scientist spesso hanno difficoltà a utilizzare direttamente questi modelli perché richiedono competenze in materia di infrastruttura. L'aggiunta di un portale per sviluppatori come Backstage offre ai data scientist un modo intuitivo per implementare ambienti ML standardizzati senza dover comprendere i dettagli dell'infrastruttura sottostante.

Utilizzando Backstage come piattaforma self-service e integrando modelli di intelligenza artificiale preconfigurati, puoi: SageMaker

- Accelera il time-to-value delle tue iniziative di machine learning.
- Contribuisci a rafforzare la sicurezza e la governance coerenti.
- Fornisci ai data scientist ambienti standardizzati e conformi.
- Riduci il sovraccarico operativo e la complessità dell'infrastruttura.

Questo modello fornisce una soluzione che affronta le sfide critiche di MLOps e fornisce anche un framework scalabile e ripetibile che consente l'innovazione mantenendo al contempo gli standard organizzativi.

Destinatari

Questo modello è destinato a un vasto pubblico coinvolto nel machine learning, nell'architettura cloud e nell'ingegneria delle piattaforme all'interno di un'organizzazione. Questo include:

- Ingegneri ML che desiderano standardizzare e automatizzare le implementazioni dei flussi di lavoro ML.
- Scienziati dei dati che desiderano accedere in modalità self-service ad ambienti ML preconfigurati e conformi.
- Ingegneri di piattaforma responsabili della creazione e della manutenzione di piattaforme di sviluppo interne e dell'infrastruttura condivisa.
- Architetti del cloud che progettano soluzioni cloud scalabili, sicure ed economiche per. MLOps
- DevOps ingegneri interessati a estendere le pratiche di integrazione continua e distribuzione continua (CI/CD) al provisioning e ai flussi di lavoro dell'infrastruttura ML.
- Responsabili tecnici e manager che supervisionano le iniziative di machine learning e desiderano migliorare la produttività, la governance e il time-to-market dei team.

Per ulteriori informazioni sulle MLOps sfide, sui MLOps moduli di SageMaker intelligenza artificiale e su come la soluzione fornita da questo modello può soddisfare le esigenze dei team di machine learning, consulta la sezione Informazioni aggiuntive.

Prerequisiti e limitazioni

Prerequisiti

- AWS Identity and Access Management (IAM) ruoli e autorizzazioni per il provisioning di risorse nel Account AWS
- Comprensione dei concetti di Amazon SageMaker Studio, SageMaker Projects, SageMaker Pipelines e SageMaker Model Registry
- Comprensione dei principi IaC ed esperienza con strumenti come AWS Cloud Development Kit (AWS CDK)

Limitazioni

- Copertura limitata dei modelli. Attualmente, la soluzione supporta solo i AIOps moduli SageMaker relativi all'intelligenza artificiale della soluzione più ampiaAIOps . Altri moduli, come Ray on Amazon Elastic Kubernetes Service (Amazon EKS), MLflow Apache Airflow e il fine-tuning per Amazon Bedrock, non sono ancora disponibili come modelli di Backstage.
- Impostazioni predefinite non configurabili. I modelli utilizzano configurazioni predefinite fisse dei AIOps SageMaker moduli senza alcuna personalizzazione. Non è possibile modificare i tipi di

istanze, le dimensioni di archiviazione, le configurazioni di rete o le politiche di sicurezza tramite l'interfaccia Backstage, il che limita la flessibilità per casi d'uso specifici.

- AWS-solo supporto. La piattaforma è progettata esclusivamente per le AWS implementazioni e non supporta scenari multicloud. Organizations che utilizzano servizi cloud esterni Cloud AWS non possono utilizzare questi modelli per le proprie esigenze di infrastruttura ML.
- Gestione manuale delle credenziali. È necessario fornire manualmente le AWS credenziali per ogni distribuzione. Questa soluzione non fornisce l'integrazione con i provider di identità aziendali o la rotazione AWS IAM Identity Center automatica delle credenziali.
- Gestione limitata del ciclo di vita. I modelli non dispongono di funzionalità complete per la gestione del ciclo di vita delle risorse, come politiche di pulizia automatizzate, consigli per l'ottimizzazione dei costi e rilevamento delle deviazioni dell'infrastruttura. È necessario gestire e monitorare manualmente le risorse distribuite dopo la creazione.

Architecture

Il diagramma seguente mostra l'architettura della soluzione per un portale per sviluppatori unificato che standardizza e accelera l'implementazione dell'infrastruttura ML con l'intelligenza artificiale in tutti gli ambienti. SageMaker

In questa architettura:

1. AWS i progetti di modernizzazione delle applicazioni forniscono la configurazione dell'infrastruttura con un cluster Amazon EKS come base per il framework Cloud Native Operational Excellence (CNOE). Questa soluzione completa affronta le complesse sfide di gestione dell'infrastruttura nativa del cloud fornendo una piattaforma di sviluppo interna scalabile (IDP). I progetti offrono un approccio strutturato alla configurazione di un'infrastruttura solida e flessibile in grado di adattarsi alle esigenze organizzative in evoluzione.
2. Il framework open source CNOE consolida DevOps gli strumenti e risolve la frammentazione dell'ecosistema attraverso un approccio ingegneristico di piattaforma unificato. Riunendo strumenti e tecnologie diversi, semplifica il complesso panorama dello sviluppo nativo del cloud, in modo che i team possano concentrarsi sull'innovazione anziché sulla gestione della toolchain. Il framework fornisce una metodologia standardizzata per la selezione, l'integrazione e la gestione degli strumenti di sviluppo.
3. Con CNOE, Backstage viene distribuito come out-of-the-box soluzione all'interno del cluster Amazon EKS. Backstage include una solida autenticazione tramite Keycloak e flussi di lavoro

di implementazione completi tramite Argo CD. Questa piattaforma integrata crea un ambiente centralizzato per la gestione dei processi di sviluppo e fornisce ai team un unico posto per accedere, implementare e monitorare la propria infrastruttura e le proprie applicazioni in più ambienti.

4. Un GitHub repository contiene modelli AIOps software preconfigurati che coprono l'intero ciclo di vita dell'IA. SageMaker Questi modelli soddisfano le esigenze critiche dell'infrastruttura ML, tra cui il provisioning di SageMaker Studio, la formazione dei modelli, le pipeline di inferenza e il monitoraggio dei modelli. Questi modelli ti aiutano ad accelerare le tue iniziative di machine learning e a garantire la coerenza tra diversi progetti e team.
5. GitHub Actions implementa un flusso di lavoro automatizzato che attiva dinamicamente l'approvvigionamento delle risorse tramite l'utilità Seed-Farmer. Questo approccio integra il catalogo Backstage con il repository dei AIOps moduli e crea un processo di implementazione dell'infrastruttura semplificato. L'automazione riduce l'intervento manuale, minimizza l'errore umano e garantisce una creazione di infrastrutture rapida e coerente in diversi ambienti.
6. Ti AWS CDKaiuta a definire e fornire l'infrastruttura sotto forma di codice e garantisce l'implementazione ripetibile, sicura e conforme delle risorse su determinate aree. Account AWS Questo approccio offre la massima governance con un intervento manuale minimo, in modo da poter creare modelli di infrastruttura standardizzati che possono essere facilmente replicati, controllati in base alla versione e verificati.

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che consente di definire e fornire l'infrastruttura nel codice. Cloud AWS
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire AWS Kubernetes senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- Amazon SageMaker AI è un servizio di machine learning gestito che ti aiuta a creare e addestrare modelli di machine learning per poi distribuirli in un ambiente ospitato pronto per la produzione.

Altri strumenti

- Backstage è un framework open source che ti aiuta a creare portali interni per sviluppatori.
- GitHub Actions è una CI/CD piattaforma che automatizza i flussi di lavoro di sviluppo software, comprese attività come la creazione, il test e la distribuzione del codice.

Archivi di codice

Questo modello utilizza codice e modelli dai seguenti GitHub repository:

- AIOps piattaforma di sviluppo interna (IDP) con repository Backstage
- SageMaker Moduli relativi all'intelligenza artificiale dal repository dei moduli AWS AIOps
- Ingegneria moderna sul repository AWS

Implementazione

Questa implementazione utilizza un modello di implementazione di livello di produzione per Backstage tratto dal Modern engineering on repository. AWS Questo approccio semplifica notevolmente il processo di configurazione incorporando al contempo le AWS migliori pratiche per la sicurezza e la scalabilità.

La sezione Epics di questo modello delinea l'approccio di implementazione. Per istruzioni dettagliate sull' step-by-stepimplementazione, consulta la guida completa all'implementazione disponibile nella piattaforma di sviluppo AIOps interna (IDP) con repository Backstage. L'implementazione include:

- Implementazione iniziale della piattaforma Backstage
- Integrazione di modelli SageMaker software con Backstage
- Utilizzo e manutenzione dei modelli di Backstage

La guida all'implementazione include anche indicazioni per la manutenzione continua, la risoluzione dei problemi e la scalabilità della piattaforma.

Best practice

Segui queste best practice per contribuire a garantire la sicurezza, la governance e l'eccellenza operativa nelle implementazioni MLOps dell'infrastruttura.

Gestione dei modelli

- Non apportare mai modifiche sostanziali ai modelli attivi.
- Verifica sempre accuratamente gli aggiornamenti prima di distribuirli in produzione.
- Mantieni versioni dei modelli chiare e ben documentate.

Sicurezza

- Associa GitHub le azioni a specifici algoritmi di commit secure hash (SHAs) per aiutare a prevenire gli attacchi alla catena di approvvigionamento.
- Utilizza ruoli IAM con privilegi minimi con autorizzazioni granulari.
- Archivia le credenziali sensibili in Secrets e. GitHub Gestione dei segreti AWS
- Non codificare mai le credenziali nei modelli.

Governance e tracciamento

- Implementa standard completi di etichettatura delle risorse.
- Abilita il monitoraggio preciso dei costi e la conformità.
- Mantieni percorsi di controllo chiari per le modifiche all'infrastruttura.

Questa guida fornisce una solida base per l'implementazione di queste best practice utilizzando i moduli Backstage, SageMaker AI e IaC.

Epiche

Configura il tuo ambiente ML

Operazione	Description	Competenze richieste
Implementa Backstage.	Questa fase utilizza i progetti del Modern Engineering on AWS repository per creare un'infrastruttura robusta e scalabile che integri più soluzioni per creare un IDP centralizzato Servizi AWS per i flussi di lavoro ML. Segui le istruzioni nella sezione relativa alla distribuzione in backstage della guida all'implementazione per clonare il repository, installare le dipendenze, avviare le variabili di ambiente di configurazione e distribui	Ingegnere della piattaforma

Operazione	Description	Competenze richieste
	re la piattaforma Backstage. AWS CDK L'infrastruttura utilizza Amazon EKS come piattaforma di orchestrazione dei container per la distribuzione di componenti IDP. L'architettura Amazon EKS include configurazioni di rete sicure per stabilire un rigoroso isolamento della rete e controllare i modelli di accesso. La piattaforma si integra con i meccanismi di autenticazione per aiutare a proteggere l'accesso degli utenti a tutti i servizi e gli ambienti.	

Operazione	Description	Competenze richieste
Configura i tuoi modelli di SageMaker intelligenza artificiale.	Questo passaggio utilizza gli script nella piattaforma di sviluppo GitHub AIOps interna (IDP) con repository Backstage. Segui le istruzioni nella sezione sulla configurazione del SageMaker modello della guida alla distribuzione per clonare il repository, impostare i prerequisiti ed eseguire lo script di installazione. Questo processo crea un repository che contiene i modelli di SageMaker intelligenza artificiale necessari per l'integrazione con Backstage.	Ingegnere della piattaforma
Integra i modelli di SageMaker intelligenza artificiale con Backstage.	Segui le istruzioni nella sezione sull'integrazione dei SageMaker modelli della guida all'implementazione per registrare i tuoi modelli di SageMaker intelligenza artificiale. Questo passaggio integra i AIOps moduli (modelli SageMaker AI dell'ultimo passaggio) nella distribuzione Backstage in modo da poter soddisfare in autonomia le esigenze dell'infrastruttura ML.	Ingegnere della piattaforma

Operazione	Description	Competenze richieste
Usa i modelli di SageMaker intelligenza artificiale di Backstage.	Segui le istruzioni nella sezione Utilizzo dei SageMaker modelli della guida alla distribuzione per accedere al portale Backstage e creare l'ambiente ML in SageMaker Studio. Nel portale Backstage, puoi scegliere tra i modelli di SageMaker intellige nza artificiale disponibi li, incluse opzioni per ambienti SageMaker Studio, SageMaker notebook, modelli di SageMaker progetto personalizzati e pipeline di distribuzione dei modelli. Dopo aver fornito i parametri di configurazione, la piattafor ma crea automaticamente repository dedicati e fornisce risorse tramite Actions e Seed-Farmer. AWS GitHub È possibile monitorare i progressi tramite i registri GitHub delle azioni e il catalogo dei componenti Backstage.	Scienziato dei dati, ingegnere dei dati, sviluppatore

Gestisci i modelli per la governance e la conformità

Operazione	Description	Competenze richieste
Aggiorna i modelli di SageMaker intelligenza artificiale.	Per aggiornare un modello di SageMaker intelligenza artificiale in Backstage, segui questi passaggi. 1. Modifica il contenuto del modello: a. Apporta le modifiche necessarie nel `template.yaml` file o modificando i file nella `skeleton/` directory. b. Testa eventuali nuovi parametri, azioni o strutture di file localment e o in un ambiente di sviluppo. 2. Modifiche al test: a. Usa l'interfaccia utente di Backstage o la CLI `@backstage/create-app` () per impalcare un component e di test utilizzando il modello aggiornato. b. Verifica che tutti i passaggi vengano eseguiti correttamente e che il codice generato soddisfi le tue aspettati ve.	Ingegnere della piattaforma

Operazione	Description	Competenze richieste
	3. Conferma e invia le modifiche: a. Invia le modifiche al repository Git in cui è archiviato il modello. Se il modello è registrato in un ramo specifico (ad esempio,`main`), gli aggiornamenti verranno riflessi automaticamente. b. Se utilizzi il controllo delle versioni (vedi il passaggio successivo), assicurati che sia aggiornata la versione o il tag corretti.	

Operazione	Description	Competenze richieste

Crea e gestisci più versioni di un modello.	Per modificare o aggiornare rapidamente, potresti voler creare più versioni di un modello di SageMaker intelligenza artificiale.
	1. Usa tag o rami Git per ogni versione; ad esempio:
	`git checkout -b v2.0.0`
	`git push origin v2.0.0`
	2. (Facoltativo ma consigliato) Registra ogni versione separatamente.
	In Backstage, puoi registrare diverse versioni di un modello come entità separate nel catalogo, dove ogni entità punta a un ramo o tag specifico. Ad esempio (per un `.yaml` file):
	`metadata:`
	`name: node-service-template-v2`
	`description: Node.js service template - Version 2`
	`spec:`
	`type: template`
	`lifecycle: experimental`
	`version: '2.0.0'`
		Ingegnere della piattaforma

Operazione	Description	Competenze richieste
	3. Comunica le modifiche in modo chiaro mantenendo un `CHANGELOG.md` file nell'archivio dei modelli. In questo file, documenta quali funzionalità o modifiche sono state introdotte in ciascuna versione del modello. 4. Se necessario, deprecate le versioni precedenti del modello: a. Contrassegnalo come obsoleto nella descrizione del modello. b. Rimuovi la versione dal catalogo se non è più necessaria.	

Estendi il tuo ambiente ML

Operazione	Description	Competenze richieste
Espandi la copertura dei modelli oltre l' SageMaker IA.	La soluzione attuale implementa solo modelli relativi all' SageMaker intelligenza artificiale AIOps . È possibile estendere l'ambiente ML aggiungendo AIOps moduli e integrando modelli software personalizzati per applicazioni aggiuntive. Servizi AWS È possibile crearli utilizzando l'interfaccia di	Ingegnere della piattaforma

Operazione	Description	Competenze richieste
	progettazione dei modelli in Backstage, implementando azioni di scaffolder personalizzate o gestendo gli archivi di modelli con metadati standard. La piattaforma supporta il controllo delle versioni dei modelli, la condivisione tra team e i flussi di lavoro di convalida per garantire la coerenza. Per ulteriori informazioni, consulta la documentazione di Backstage. Puoi anche implementare modelli di ereditarietà dei modelli per creare versioni specializzate dei modelli di base. Questa estensibilità consente di gestire diverse AWS risorse e applicazioni oltre all' SageMaker intelligenza artificiale, preservando al contempo l'esperienza semplificata degli sviluppatori e mantenendo gli standard dell'organizzazione.	

Operazione	Description	Competenze richieste
Usa l'iniezione dinamica dei parametri.	I modelli attuali utilizzano configurazioni predefinite senza personalizzazione ed eseguono la CLI Seed-Farmer per distribuire risorse con variabili predefinite. È possibile estendere la configurazione predefinita utilizzando l'iniezione dinamica dei parametri per configura zioni specifiche del modulo.	Ingegnere della piattaforma
Migliora la sicurezza e la conformità.	Per migliorare la sicurezza nella creazione di risorse AWS, puoi abilitare l'integra zione del controllo degli accessi basato sul ruolo (RBAC) con single sign-on (SSO), SAML, OpenID Connect (OIDC) e policy come applicazione del codice.	Ingegnere della piattaforma
Aggiungi la pulizia automatica delle risorse.	È possibile abilitare funzional ità per le politiche di pulizia automatizzate e aggiunger e anche il rilevamento e la correzione delle deviazioni dell'infrastruttura.	Ingegnere della piattaforma

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Rimuovi l'infrastruttura Backstage e le risorse SageMaker AI.	Quando hai finito di utilizzar e l'ambiente ML, segui le istruzioni nella sezione Pulizia e gestione delle risorse della guida all'implementazione per rimuovere l'infrastruttura Backstage ed eliminare le risorse di SageMaker intellige nza artificiale nel tuo ambiente ML.	Ingegnere della piattaforma

risoluzione dei problemi

Problema	Soluzione
AWS CDK errori di bootstrap	Verifica AWS le credenziali e la configurazione della regione.
Problemi di accesso al cluster Amazon EKS	Controlla la configurazione di kubectl e le autorizzazioni IAM.
Problemi di connettività con Application Load Balancer	Assicurati che i gruppi di sicurezza consentano il traffico in entrata sulla porta 80/443.
GitHub problemi di integrazione	Verifica le autorizzazioni dei GitHub token e l'accesso all'organizzazione.
SageMaker Errori di implementazione dell'IA	Controlla le Servizio AWS quote e le autorizza zioni IAM.

Risorse correlate

- Ingegneria della piattaforma (nella guida AWS Cloud Adoption Framework: Platform perspective)

- Documentazione Amazon SageMaker AI
- Modelli di software Backstage (sito Web Backstage)
- AIOps repository di moduli (raccolta di moduli IAc riutilizzabili per ML)
- AIOps piattaforma di sviluppo interna (IDP) con repository Backstage
- Ingegneria moderna sul repository AWS
- Sito Web Cloud Native Operational Excellence (CNOE)

Informazioni aggiuntive

Sfide aziendali

Organizations che intraprendono o ampliano le proprie MLOps iniziative spesso incontrano queste sfide aziendali e tecniche:

- Ambienti incoerenti. La mancanza di ambienti di sviluppo e implementazione standardizzati rende difficile la collaborazione e aumenta i rischi di implementazione.
- Sovraccarico di approvvigionamento manuale. La configurazione manuale di un'infrastruttura ML con SageMaker Studio, bucket Amazon Simple Storage Service (Amazon S3), ruoli IAM CI/CD e pipeline richiede molto tempo ed è soggetta a errori e distoglie i data scientist dal loro compito principale di sviluppo di modelli.
- Mancanza di reperibilità e riutilizzo. La mancanza di un catalogo centralizzato rende difficile trovare modelli ML, set di dati e pipeline esistenti. Ciò comporta un lavoro ridondante e la perdita di opportunità di riutilizzo.
- Governance e conformità complesse. Garantire che i progetti di machine learning aderiscano alle politiche di sicurezza organizzative, alle normative sulla privacy dei dati e agli standard di conformità come l'Health Insurance Portability and Accountability Act (HIPAA) e il Regolamento generale sulla protezione dei dati (GDPR) può essere difficile senza barriere automatizzate.
- Tempi di valorizzazione lenti. L'effetto cumulativo di queste sfide si traduce in un prolungamento dei cicli di vita dei progetti di machine learning e ritarda la realizzazione del valore aziendale derivante dagli investimenti in machine learning.
- Rischi per la sicurezza. Configurazioni incoerenti e processi manuali possono introdurre vulnerabilità di sicurezza che rendono difficile l'applicazione dei privilegi minimi e l'isolamento della rete.

Questi problemi prolungano i cicli di sviluppo, aumentano il sovraccarico operativo e introducono rischi per la sicurezza. La natura iterativa del machine learning richiede flussi di lavoro ripetibili e una collaborazione efficiente.

Gartner prevede che entro il 2026, l'80% delle organizzazioni di ingegneria del software disporrà di team di piattaforma. (Vedi Platform Engineering consente agli sviluppatori di essere migliori, più veloci e più felici sul sito Web di Gartner.) Questa previsione evidenzia come un IDP possa accelerare la distribuzione del software. In qualità di IDP, Backstage aiuta a ripristinare l'ordine in un'infrastruttura complessa in modo che i team possano fornire codice di alta qualità in modo rapido e sicuro. L'integrazione di Backstage con AIOps moduli rinforzati ti aiuta a passare dalla risoluzione dei problemi reattiva alla prevenzione proattiva.

MLOps SageMaker moduli

I AIOps moduli del GitHub repository utilizzati per questo modello forniscono una base preziosa per la standardizzazione MLOps AWS tramite IaC riutilizzabili e rinforzati. Questi moduli racchiudono le migliori pratiche per il provisioning di SageMaker progetti, pipeline e risorse di rete e storage associate, con l'obiettivo di ridurre la complessità e accelerare la configurazione degli ambienti ML. È possibile utilizzare questi modelli per vari casi MLOps d'uso per stabilire modelli di distribuzione coerenti e sicuri che promuovano un approccio più governato ed efficiente ai flussi di lavoro ML.

L'utilizzo diretto AIOps dei moduli spesso richiede ai team della piattaforma di implementare e gestire questi modelli IaC, il che può rappresentare una sfida per i data scientist che desiderano un accesso self-service. La scoperta e la comprensione dei modelli disponibili, la configurazione dei parametri necessari e l'attivazione della loro implementazione potrebbero richiedere l'utilizzo di Servizio AWS console di navigazione o l'interazione diretta con gli strumenti IaC. Ciò può creare attriti, aumentare il carico cognitivo per i data scientist che preferiscono concentrarsi sulle attività di machine learning e potenzialmente portare a parametrizzazioni incoerenti o deviazioni dagli standard organizzativi se questi modelli non sono gestiti tramite un'interfaccia centralizzata e intuitiva. L'integrazione di questi potenti AIOps moduli con un IDP come Backstage aiuta ad affrontare queste sfide fornendo un'esperienza self-service semplificata, una maggiore reperibilità e controlli di governance più solidi per l'utilizzo di questi elementi costitutivi standardizzati. MLOps

Backstage come IDP

Una piattaforma di sviluppo interna (IDP) è un livello self-service creato dai team di piattaforma per semplificare e standardizzare il modo in cui gli sviluppatori creano, distribuiscono e gestiscono le applicazioni. Riduce la complessità dell'infrastruttura e offre agli sviluppatori un facile accesso a strumenti, ambienti e servizi tramite un'interfaccia unificata.

L'obiettivo principale di un IDP è migliorare l'esperienza e la produttività degli sviluppatori mediante:

- Abilitare il self-service per attività come la creazione e l'implementazione di servizi.
- Promuovere la coerenza e la conformità tramite modelli standard.
- Integrazione degli strumenti durante l'intero ciclo di vita dello sviluppo (CI/CD, monitoraggio e documentazione).

Backstage è un portale per sviluppatori open source creato da Spotify e ora parte della Cloud Native Computing Foundation (CNCF). Aiuta le organizzazioni a creare il proprio IDP fornendo una piattaforma centralizzata ed estensibile per gestire componenti software, strumenti e documentazione. Con Backstage, gli sviluppatori possono:

- Scopri e gestisci tutti i servizi interni tramite un catalogo software.
- Crea nuovi progetti utilizzando modelli predefiniti tramite il plug-in scaffolder.
- Accedi a strumenti integrati come CI/CD pipeline, dashboard Kubernetes e sistemi di monitoraggio da un'unica posizione.
- Mantieni una documentazione coerente e basata su markdown tramite. TechDocs

DOMANDE FREQUENTI

Qual è la differenza tra l'utilizzo di questo modello di Backstage e la distribuzione manuale di SageMaker Studio tramite la console? SageMaker

Il modello Backstage offre diversi vantaggi rispetto all'implementazione manuale della AWS console, tra cui configurazioni standardizzate che seguono le migliori pratiche organizzative, l'implementazione automatizzata di IaC tramite Seed-Farmer e le politiche di sicurezza e misure di conformità integrate e l' AWS CDK integrazione con i flussi di lavoro degli sviluppatori dell'organizzazione. GitHub Il modello crea anche implementazioni riproducibili con controllo delle versioni, che semplificano la replica degli ambienti nelle diverse fasi (sviluppo, gestione temporanea, produzione) e mantengono la coerenza tra i team. Inoltre, il modello include funzionalità di pulizia automatizzate e si integra con il sistema di gestione delle identità dell'organizzazione tramite Backstage. L'implementazione manuale tramite la console richiede AWS competenze approfondite e non fornisce il controllo della versione o lo stesso livello di standardizzazione e governance offerto dal modello. Per questi motivi, le implementazioni su console sono più adatte per esperimenti una tantum rispetto agli ambienti ML di produzione.

Cos'è Seed-Farmer e perché questa soluzione lo utilizza?

Seed-Farmer è uno strumento di orchestrazione AWS dell'implementazione che gestisce i moduli dell'infrastruttura utilizzando il. AWS CDK Questo modello utilizza Seed-Farmer perché fornisce componenti di infrastruttura standardizzati e riutilizzabili progettati specificamente per i AI/ML carichi di lavoro, gestisce automaticamente dipendenze complesse tra loro e garantisce implementazioni coerenti in diversi ambienti. Servizi AWS

Devo installare il per utilizzare questi modelli? AWS CLI

No, non è necessario installarli AWS CLI sul computer. I modelli funzionano interamente tramite GitHub Actions nel cloud. Fornisci AWS le tue credenziali (chiave di accesso, chiave segreta e token di sessione) tramite l'interfaccia Backstage e la distribuzione avviene automaticamente nell'ambiente GitHub Actions.

Quanto tempo è necessario per implementare un SageMaker ambiente Studio?

Il completamento di una tipica implementazione di SageMaker Studio richiede 15-25 minuti. Ciò include il AWS CDK bootstrap (2-3 minuti), la configurazione della toolchain Seed-Farmer (3-5 minuti) e la creazione di risorse (10-15 minuti). L'ora esatta dipende dalla vostra configurazione Regione AWS e dalla complessità della vostra configurazione di rete.

Posso implementare più SageMaker ambienti nello stesso Account AWS ambiente?

Sì, è possibile. Ogni distribuzione crea risorse con nomi univoci in base al nome del componente fornito nel modello. Tuttavia, fai attenzione alle Servizio AWS quote: ogni account può avere un numero limitato di SageMaker domini per regione, quindi controlla le quote prima di creare più ambienti.

Automatizza le operazioni AWS dell'infrastruttura utilizzando Amazon Bedrock

Ishwar Chauthaiwale e Anand Bukkapatnam Tirumala, Amazon Web Services

Riepilogo

Nelle soluzioni native per il cloud, l'automazione delle operazioni delle infrastrutture comuni svolge un ruolo fondamentale nel mantenimento di ambienti efficienti, sicuri ed economici. La gestione manuale delle operazioni richiede molto tempo ed è soggetta a errori umani. Inoltre, i membri del team con diversi livelli di AWS esperienza devono svolgere queste attività garantendo al contempo la conformità ai protocolli di sicurezza. Questo modello dimostra come utilizzare Amazon Bedrock per automatizzare le operazioni di AWS infrastruttura comuni tramite l'elaborazione del linguaggio naturale (NLP).

Questo modello può aiutare le organizzazioni a sviluppare codice riutilizzabile, modulare e sicuro per la distribuzione di infrastrutture generative basate sull'intelligenza artificiale in più ambienti. Grazie alla sua attenzione all'infrastruttura come codice (IaC) e all'automazione, offre DevOps vantaggi chiave tra cui controllo delle versioni, implementazioni coerenti, riduzione degli errori, provisioning più rapido e collaborazione migliorata.

Il modello implementa un'architettura sicura che consente ai team di gestire le operazioni relative a fattori chiave, tra cui: Servizi AWS

- Gestione delle versioni dei bucket Amazon Simple Storage Service (Amazon S3)
- Creazione di snapshot di Amazon Relational Database Service (Amazon RDS)
- Gestione delle istanze Amazon Elastic Compute Cloud (Amazon EC2)

L'architettura utilizza endpoint Amazon Virtual Private Cloud (Amazon VPC) e reti private per comunicazioni sicure, AWS Lambda con funzioni che funzionano come esecutori di attività all'interno di sottoreti private. Amazon S3 fornisce la gestione dei dati e implementa ruoli e autorizzazioni completi AWS Identity and Access Management (IAM) per garantire controlli di accesso adeguati. Questa soluzione non include una funzionalità di cronologia chat e la chat non viene archiviata.

Prerequisiti e limitazioni

- Un attivo Account AWS.

- È necessario adottare adeguate misure di controllo degli accessi per garantire e controllare l'accesso. Esempi di controllo degli accessi includono l'utilizzo di modelli di base AWS Systems Manager, un ruolo IAM per la distribuzione e ruoli basati sui servizi, la disabilitazione dell'accesso pubblico ai bucket Amazon S3 e la configurazione di una coda di lettere morte.
- Una chiave () gestita dal cliente. AWS Key Management ServiceAWS KMS
- AWS Command Line Interface (AWS CLI) versione 2 o successiva, installata e configurata nell'ambiente di distribuzione.
- Terraform AWS Provider versione 4 o successiva installato e configurato.
- Terraform versione 1.5.7 o successiva installata e configurata.
- Rivedi e definisci gli schemi OpenAPI per i gruppi di azione dei tuoi agenti in Amazon Bedrock per proteggerti da accessi non autorizzati e mantenere l'integrità dei dati.
- Accesso abilitato nel tuo Account AWS per i modelli Amazon Titan Text Embeddings v2 richiesti e per i modelli Claude 3.5 Sonnet o Claude 3 Haiku Foundation. Per evitare errori di distribuzione, verifica che la distribuzione di destinazione supporti i modelli richiesti. Regione AWS
- Un cloud privato virtuale (VPC) configurato che segue le best practice di AWS Well Architected Framework.
- Revisione completata della politica di Amazon Responsible AI.

Versioni del prodotto

- Incorporamenti di testo Amazon Titan v2
- Anthropic Claude 3.5 Sonnet o Claude 3 Haiku
- Terraform Provider versione 4 o successiva AWS
- Terraform versione 1.5.7 o successiva

Architecture

Il diagramma seguente mostra i componenti del flusso di lavoro e dell'architettura per questo modello.

L'architettura della soluzione è composta da più livelli che collaborano per elaborare le richieste in linguaggio naturale ed eseguire le operazioni corrispondenti AWS :

1. L'utente effettua richieste operative tramite la console di chat di Amazon Bedrock.

2. Il chatbot utilizza Amazon Bedrock Knowledge Bases per l'elaborazione delle richieste. Implementa il modello Amazon Titan Text Embeddings v2 per l'elaborazione del linguaggio naturale.
3. Se il prompt dell'utente include una richiesta di azione, il gruppo di azioni Amazon Bedrock utilizza il modello Anthropic Claude 3 Haiku o il modello Claude 3.5 Sonnet (a seconda della scelta) per la logica di esecuzione e definisce le operazioni tramite uno schema OpenAPI.
4. Il gruppo di azione raggiunge gli endpoint Amazon VPC utilizzando AWS PrivateLink per una comunicazione di servizio sicura.
5. La AWS Lambda funzione viene raggiunta tramite gli endpoint Amazon VPC per i servizi Amazon Bedrock.
6. Le funzioni Lambda sono il motore di esecuzione principale. In base alla richiesta, la funzione Lambda chiama l'API per eseguire azioni su. Servizi AWS La funzione Lambda gestisce anche il routing e l'esecuzione delle operazioni.
7. Vengono eseguite la richiesta Servizi AWS get the API dalla funzione Lambda e le operazioni corrispondenti.
8. La funzione Lambda calcola un payload di output compreso da Amazon Bedrock.
9. Questo payload viene inviato ad Amazon Bedrock utilizzando un servizio PrivateLink di comunicazione sicuro. Il modello di linguaggio di grandi dimensioni (LLM) utilizzato da Amazon Bedrock comprende questo payload e lo converte in un formato comprensibile per l'uomo.
10. L'output viene quindi mostrato all'utente sulla console di chat di Amazon Bedrock.

La soluzione consente le seguenti operazioni principali:

- Amazon S3: abilita il controllo delle versioni dei bucket per il controllo delle versioni.
- Amazon RDS: crea snapshot del database per il backup.
- Amazon EC2: elenca le istanze e controlla l'avvio e l'arresto delle istanze.

Tools (Strumenti)

Servizi AWS

- Amazon Bedrock è un servizio completamente gestito che rende disponibili per l'uso modelli di base ad alte prestazioni (FMs) delle principali startup di intelligenza artificiale e di Amazon tramite un'API unificata.

- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire Servizi AWS tramite comandi nella shell della riga di comando.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel Cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon OpenSearch Serverless è una configurazione serverless su richiesta per Amazon Service. OpenSearch
- AWS PrivateLinkti aiuta a creare connessioni private unidirezionali dai tuoi cloud privati virtuali (VPCs) a servizi esterni al VPC.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale in. Cloud AWS
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Systems Managerti aiuta a gestire le applicazioni e l'infrastruttura in esecuzione in. Cloud AWS Semplifica la gestione delle applicazioni e delle risorse, riduce i tempi di rilevamento e risoluzione dei problemi operativi e aiuta a gestire le AWS risorse in modo sicuro su larga scala.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.

Altri strumenti

- Git è un sistema di controllo delle versioni distribuito e open source.
- Terraform è uno strumento di infrastruttura come codice (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.

Archivio di codici

Il codice per questo pattern è disponibile nel repository GitHub infra-ops-orchestratoraws-samples/.

Best practice

- Monitora regolarmente i log di esecuzione di Lambda. Per ulteriori informazioni, consulta Monitoraggio e risoluzione dei problemi delle funzioni Lambda. Per ulteriori informazioni sulle best practice, consulta Best practice for working with AWS Lambda functions.
- Rivedi periodicamente le configurazioni di sicurezza per garantire la conformità ai requisiti dell'organizzazione. Per ulteriori informazioni, consulta Best practice in materia di sicurezza.
- Segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.

Epiche

Implementazione della soluzione

Operazione	Description	Competenze richieste

Clonare il repository.	Per clonare il repository sul computer locale, esegui il seguente comando:
	`git clone "git@github.com:aws-samples/infra-ops-orchestrator.git"`
	`cd infra-ops-orchestrator`
		AWS DevOps, DevOps ingegnere

Modifica le variabili di ambiente.	Modifica il `terraform.tfvars` file. Nella directory principale del repository clonato. Esamina i segnaposti indicati da `[XXXXX]` e aggiornali in base al tuo ambiente.	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste

Crea l'infrastruttura.	Per creare l'infrastruttura, esegui i seguenti comandi:
	`terraform init`
	`terraform plan`
	Esamina attentamente il piano di esecuzione. Se le modifiche pianificate sono accettabili, esegui il seguente comando:
	`terraform apply --auto-approve`
		AWS DevOps, DevOps ingegnere

Accedi alla soluzione

Operazione	Description	Competenze richieste
Accedi alla soluzione.	Dopo una corretta implement azione, segui questi passaggi per utilizzare l'interfaccia basata sulla chat: 1. Per accedere all'Infra structure Orchestrator Assistant, accedi al ruolo Console di gestione AWS Using an IAM con autorizza zioni Amazon Bedrock e apri la console Amazon Bedrock all'indirizzo. https://console.aws.amazon.com/bedrock/ Seleziona Agenti	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	nel riquadro di navigazione a sinistra. Quindi, scegli Infrastructure Orchestrator Assistant nella sezione Agenti. 2. Assicurati che nel tuo AWS ambiente esistano risorse di destinazione per i seguenti suggerimenti, quindi prova queste operazioni di esempio: • Crea un backup istantaneo di un'istanza Amazon RDS chiedendo: 'Crea uno snapshot dell'istanza RDS [nome-istanza] ' • Abilita il controllo delle versioni su un bucket Amazon S3 chiedendo: 'Abilita il controllo delle versioni per il bucket [bucket-name] ' • Elenca le istanze Amazon EC2 chiedendo: 'Elenca tutte le istanze EC2' • Avvia o arresta un'istanza Amazon EC2 chiedendo: 'Start EC2 instance [instance-id] 'o 'Stop EC2 instance [instance-id]' Nota: sostituisci i valori tra parentesi con i nomi delle	

Operazione	Description	Competenze richieste
	risorse effettivi o provenienti dal tuo ambiente. IDs AWS	

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste

Eliminare le risorse create.	Per eliminare tutta l'infrast ruttura creata da questo modello, esegui il seguente comando:
	`terraform plan -destroy`
	Esamina attentamente il piano di distruzione. Se le eliminazi oni pianificate sono accettabili, esegui il seguente comando:
	`terraform destroy`
	Nota: questo comando eliminerà definitivamente tutte le risorse create da questo modello. Il comando richieder à una conferma prima di rimuovere qualsiasi risorsa.
		AWS DevOps, DevOps ingegnere

Risoluzione dei problemi

Problema	Soluzione
Comportamento dell'agente	Per informazioni su questo problema, consulta Test e risoluzione dei problemi del comportam

Problema	Soluzione
	ento degli agenti nella documentazione di Amazon Bedrock.
Problemi relativi alla rete Lambda	Per informazioni su questi problemi, consulta Risoluzione dei problemi di rete in Lambda nella documentazione di Lambda.
autorizzazioni IAM	Per informazioni su questi problemi, consulta Risoluzione dei problemi di IAM nella documentazione IAM.

Risorse correlate

- Creazione di uno snapshot DB per un'istanza DB Single-AZ per Amazon RDS
- Definisci schemi OpenAPI per i gruppi d'azione dei tuoi agenti in Amazon Bedrock
- Abilitazione del controllo delle versioni sui bucket
- Come funziona Amazon Bedrock Agents
- Recupera dati e genera risposte AI con Amazon Bedrock Knowledge Bases
- Accedi in modo sicuro a Services Over AWS PrivateLink
- Arresta e avvia le istanze Amazon EC2
- Utilizza i gruppi di azione per definire le azioni da eseguire per il tuo agente

Automatizza CloudFront gli aggiornamenti quando gli endpoint del bilanciamento del carico cambiano utilizzando Terraform

Tamil Selvan P, Mohan Annam e Naveen Suthar, Amazon Web Services

Riepilogo

Quando gli utenti di Amazon Elastic Kubernetes Service (Amazon EKS) eliminano e reinstallano la configurazione di ingresso tramite grafici Helm, viene creato un nuovo Application Load Balancer (ALB). Ciò crea un problema perché Amazon CloudFront continua a fare riferimento al vecchio record DNS di ALB. Di conseguenza, i servizi destinati a questo endpoint non saranno raggiungibili. (Per ulteriori dettagli su questo flusso di lavoro problematico, consulta Informazioni aggiuntive.)

Per risolvere questo problema, questo schema descrive l'utilizzo di una AWS Lambda funzione personalizzata sviluppata con Python. Questa funzione Lambda rileva automaticamente quando viene creato un nuovo ALB tramite le regole di Amazon. EventBridge Utilizzando AWS SDK per Python (Boto3), la funzione aggiorna quindi la CloudFront configurazione con il nuovo indirizzo DNS di ALB, assicurando che il traffico venga indirizzato all'endpoint corretto.

Questa soluzione automatizzata mantiene la continuità del servizio senza routing o latenza aggiuntivi. Il processo aiuta a garantire che faccia CloudFront sempre riferimento all'endpoint DNS ALB corretto, anche quando l'infrastruttura sottostante cambia.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Un'applicazione Web di esempio per test e convalida che viene distribuita su Amazon EKS utilizzando Helm. Per ulteriori informazioni, consulta Distribuire applicazioni con Helm su Amazon EKS nella documentazione di Amazon EKS.
- Configura CloudFront per indirizzare le chiamate a un ALB creato da un controller di ingresso Helm. Per ulteriori informazioni, consulta Install AWS Load Balancer Controller with Helm nella documentazione di Amazon EKS e Limita l'accesso agli Application Load Balancers nella documentazione. CloudFront
- Terraform installato e configurato in uno spazio di lavoro locale.

Limitazioni

- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità regionale, vedi AWS Servizi per regione. Per endpoint specifici, consulta Service endpoints and quotas e scegli il link relativo al servizio.

Versioni del prodotto

- Terraform versione 1.0.0 o successiva
- Terraform AWS Provider versione 4.20 o successiva

Architecture

Il diagramma seguente mostra i componenti del flusso di lavoro e dell'architettura per questo modello.

Questa soluzione esegue le seguenti operazioni:

1. Il controller di ingresso Amazon EKS crea un nuovo Application Load Balancer (ALB) ogni volta che si verifica un riavvio o una distribuzione di Helm.
2. EventBridge cerca gli eventi di creazione di ALB.
3. L'evento di creazione ALB attiva la funzione Lambda.
4. La funzione Lambda è stata implementata sulla base di python 3.9 e utilizza l'API boto3 per la chiamata. Servizi AWS La funzione Lambda aggiorna la CloudFront voce con il nome DNS del load balancer più recente, ricevuto dagli eventi create load balancer.

Tools (Strumenti)

Servizi AWS

- Amazon CloudFront accelera la distribuzione dei tuoi contenuti web distribuendoli attraverso una rete mondiale di data center, che riduce la latenza e migliora le prestazioni.
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire AWS Kubernetes senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altro modo. Account AWS

- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS SDK per Python (Boto3)è un kit di sviluppo software che ti aiuta a integrare la tua applicazione, libreria o script Python con. Servizi AWS

Altri strumenti

- Python è un linguaggio di programmazione per computer generico.
- Terraform è uno strumento Infrastructure as Code (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.

Archivio di codici

Il codice per questo pattern è disponibile nel repository GitHub aws-cloudfront-automation-terraform-samples.

Epiche

Configura una workstation locale

Operazione	Description	Competenze richieste
Imposta e configura la Git CLI.	Per installare e configurare l'interfaccia a riga di comando Git (CLI) nella workstation locale, segui le istruzioni Getting Started — Installing Git nella documentazione Git.	DevOps ingegnere
Crea la cartella del progetto e aggiungi i file.	1. Vai al GitHub repository del pattern e scegli il pulsante Codice. 2. Nella finestra di dialogo Clone, scegli la scheda HTTPS. In Clona utilizzan	DevOps ingegnere

Operazione	Description	Competenze richieste
	do l'URL web, copia l'URL visualizzato. 3. Crea una cartella sul tuo computer locale. Assegna un nome con il nome del tuo progetto. 4. Apri un terminale sul tuo computer locale e accedi a questa cartella. 5. Per clonare il repository git di questo pattern, esegui il seguente comando: `git clone https://github.com/aws-samples/aws-cloudfront-automation-terraform-samples` 6. Dopo che il repository è stato clonato, usa il seguente comando per andare alla directory clonata: `cd <directory name>/cloudfront-update` Apri questo progetto in un ambiente di sviluppo integrato (IDE) a tua scelta.	

Fornisci l'architettura di destinazione utilizzando la configurazione Terraform

Operazione	Description	Competenze richieste

Distribuire la soluzione.	Per distribuire risorse nell'obiettivo Account AWS, utilizza i seguenti passaggi:
	1. Passare alla cartella `cloudfront-update` .
	2. Aggiorna il `terraform.tfvars` file con. `cloudfront_distribution_id`
	3. Per impostare il Regione AWS tuo AWS profilo, esegui il seguente comando:
	`export AWS_REGION={{ REGION }}`
	4. Per inizializzare Terraform , esegui il seguente comando:
	`terraform init`
	5. Per convalidare Terraform , esegui il seguente comando:
	`terraform validate`
	6. Per creare un piano di esecuzione Terraform , esegui il seguente comando:
		DevOps ingegnere

Operazione	Description	Competenze richieste

	`terraform plan`
	7. Per applicare le azioni `daterraform plan`, esegui il seguente comando:
	`terraform apply`	

Verifica della distribuzione

Operazione	Description	Competenze richieste
Convalida la distribuzione.	1. Accedi a Console di gestione AWS e apri la CloudFront console Amazon su https://console.aws.amazon.com/cloudfront/v4/home. 2. Nel riquadro di navigazione a sinistra, scegli Distribuzioni, quindi apri la distribuzione. CloudFront 3. Nella scheda Origini, verifica che il nome e la mappatura di origine contengano il record DNS ALB aggiornato.	DevOps ingegnere

Pulisci l'infrastruttura

Operazione	Description	Competenze richieste
Pulisci l'infrastruttura.	Per ripulire l'infrastruttura creata in precedenza, procedi nel seguente modo: 1. Esegui il comando riportato qui di seguito: `terraform destroy` 2. Per confermare il comando destroy, inserisci `yes`.	DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
Errore durante la convalida delle credenziali del provider	Quando esegui Terraform `apply` o `destroy` i comandi dal tuo computer locale, potresti riscontrare un errore simile al seguente: `Error: configuring Terraform AWS Provider: error validating provider credentials: error calling sts:GetCallerIdentity: operation error STS: GetCallerIdentity, https response error StatusCode: 403, RequestID: 123456a9-fbc1-40ed-b8d8-513d0133ba7f, api error InvalidClientTokenId: The security token included in the request is invalid.` Questo errore è causato dalla scadenza del token di sicurezza per le credenziali utilizzate nella configurazione del computer locale.

Problema	Soluzione
	Per risolvere l'errore, consulta Impostare e visualizzare le impostazioni di configurazione nella documentazione AWS Command Line Interface (AWS CLI).

Risorse correlate

AWS resources

- Limita l'accesso agli Application Load Balancer
- Indirizza il traffico Internet con AWS Load Balancer Controller

Documentazione Terraform

- AWS Provider
- Installa Terraform
- Stato remoto

Informazioni aggiuntive

Flusso di lavoro problematico

Il diagramma mostra il flusso di lavoro seguente:

1. Quando l'utente accede all'applicazione, la chiamata va a. CloudFront
2. CloudFront indirizza le chiamate al rispettivo Application Load Balancer (ALB).
3. L'ALB include gli indirizzi IP di destinazione, che sono gli indirizzi IP dell'application pod. Da lì, l'ALB fornisce all'utente i risultati attesi.

Tuttavia, questo flusso di lavoro presenta un problema. Le distribuzioni delle applicazioni avvengono tramite grafici Helm. Ogni volta che viene effettuata una distribuzione o se qualcuno riavvia Helm, viene ricreato anche il rispettivo ingresso. Di conseguenza, il controller di bilanciamento del carico esterno ricrea l'ALB. Inoltre, durante ogni ricreazione, l'ALB viene ricreato con un nome DNS diverso.

Per questo motivo, CloudFront avrà una voce obsoleta nelle impostazioni di origine. A causa di questa voce non aggiornata, l'applicazione non sarà raggiungibile dall'utente. Questo problema comporta tempi di inattività per gli utenti.

Soluzione alternativa

Un'altra soluzione possibile è creare un DNS esterno per l'ALB e quindi indirizzarlo all'endpoint della zona ospitata privata Amazon Route 53. CloudFront Tuttavia, questo approccio aggiunge un ulteriore salto nel flusso dell'applicazione, che potrebbe causare latenza dell'applicazione. La soluzione della funzione Lambda di questo pattern non interrompe il flusso di corrente.

Automatizza le CodeGuru revisioni Amazon per le applicazioni AWS CDK Python utilizzando Actions GitHub

Vanitha Dontireddy e Sarat Chandra Pothula, Amazon Web Services

Riepilogo

Nota: a partire dal 7 novembre 2025, non puoi creare nuove associazioni di repository in Amazon CodeGuru Reviewer. Per informazioni sui servizi con funzionalità simili a CodeGuru Reviewer, consulta la modifica della disponibilità di Amazon CodeGuru Reviewer nella documentazione di CodeGuru Reviewer.

Questo modello mostra l'integrazione delle revisioni CodeGuru automatiche del codice di Amazon per le applicazioni AWS Cloud Development Kit (AWS CDK) Python, GitHub orchestrate tramite Actions. La soluzione implementa un'architettura serverless definita in Python AWS CDK . Automatizzando l'analisi del codice da parte di esperti all'interno della pipeline di sviluppo, questo approccio può fare quanto segue per i progetti Python AWS CDK :

- Migliora la qualità del codice.
- Semplifica i flussi di lavoro.
- Massimizza i vantaggi dell'elaborazione senza server.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- AWS Command Line Interface (AWS CLI) versione 2.9.11 o successiva, installata e configurata.
- Un GitHub account attivo e un GitHub repository con autorizzazioni di lettura e scrittura per il flusso di lavoro e creazione di richieste pull (PR) da parte di GitHub Actions per garantire il corretto funzionamento del flusso di lavoro PR.
- Un ruolo OpenID Connect (OIDC) in GitHub Actions per implementare la soluzione in. Account AWSPer creare il ruolo, usa il costrutto.AWS CDK

Limitazioni

- Amazon CodeGuru Profiler supporta applicazioni scritte in tutti i linguaggi Java Virtual Machine (JVM) (come Scala e Kotlin) e runtime e Python 3.6 o versioni successive.
- Amazon CodeGuru Reviewer supporta le associazioni con gli archivi di codice Java e Python solo dai seguenti provider di sorgenti AWS CodeCommit: Bitbucket GitHub GitHub , Enterprise Cloud ed Enterprise Server. GitHub Inoltre, i repository Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) sono supportati solo tramite Actions. GitHub
- Non esiste un modo automatico per stampare i risultati durante la pipeline di integrazione e distribuzione continua (CI/CD). Invece, questo modello utilizza GitHub Actions come metodo alternativo per gestire e visualizzare i risultati.
- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità regionale, consulta i servizi AWS per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Architecture

Il diagramma seguente mostra l'architettura di questa soluzione.

Come mostrato nel diagramma, quando uno sviluppatore crea una pull request (PR) per la revisione, GitHub Actions attiva i seguenti passaggi:

1. Assunzione del ruolo IAM: la pipeline utilizza il ruolo IAM specificato in GitHub Secrets per eseguire le attività di distribuzione.
2. Analisi del codice
   - CodeGuru Reviewer analizza il codice memorizzato nel bucket Amazon S3. Identifica i difetti e fornisce consigli per correzioni e ottimizzazioni.
   - CodeGuru Scansioni di sicurezza alla ricerca di violazioni e vulnerabilità delle policy.
3. Revisione dei risultati
   - La pipeline stampa un collegamento alla dashboard dei risultati nell'output della console.
   - Se vengono rilevati risultati critici, la pipeline fallisce immediatamente.
   - Per risultati di elevata, normale o bassa gravità, la pipeline continua con la fase successiva.
4. Approvazione PR
   - Un revisore deve approvare manualmente il PR.
   - Se il PR viene negato, la pipeline fallisce e interrompe le ulteriori fasi di implementazione.

5. Implementazione CDK: dopo l'approvazione delle pubbliche relazioni, inizia il processo di implementazione del CDK. Imposta quanto segue Servizi AWS e le risorse:
   - CodeGuru Profiler
   - AWS Lambda funzione
   - Coda Amazon Simple Queue Service (Amazon SQS)
6. Generazione di dati di profilazione — Per generare dati di profilazione sufficienti per CodeGuru Profiler:
   - La pipeline richiama la funzione Lambda più volte inviando periodicamente messaggi alla coda di Amazon SQS.

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che ti aiuta a definire e fornire l'infrastruttura nel codice. Cloud AWS
- CDK Toolkit è un kit di sviluppo cloud a riga di comando che ti aiuta a interagire con la tua AWS CDK app.
- Amazon CodeGuru Profiler raccoglie dati sulle prestazioni di runtime dalle tue applicazioni live e fornisce consigli che possono aiutarti a ottimizzare le prestazioni delle tue applicazioni.
- Amazon CodeGuru Reviewer utilizza l'analisi dei programmi e l'apprendimento automatico per rilevare potenziali difetti difficili da individuare per gli sviluppatori. Quindi, CodeGuru Profiler offre suggerimenti per migliorare il codice Java e Python.
- Amazon CodeGuru Security è uno strumento statico per la sicurezza delle applicazioni che utilizza l'apprendimento automatico per rilevare violazioni e vulnerabilità delle policy di sicurezza. Fornisce suggerimenti per affrontare i rischi per la sicurezza e genera parametri che ti consentono di monitorare il livello di sicurezza delle tue applicazioni.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Simple Queue Service (Amazon SQS) fornisce una coda ospitata sicura, durevole e disponibile che ti aiuta a integrare e disaccoppiare sistemi e componenti software distribuiti.

- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri strumenti

- GitHub Actions è una piattaforma di integrazione e distribuzione continua (CI/CD) strettamente integrata con i repository. GitHub Puoi utilizzare GitHub Actions per automatizzare la pipeline di compilazione, test e distribuzione.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub amazon-codeguru-suite-cdk-python.

Best practice

- Aderisci alle migliori pratiche per lo sviluppo e l'implementazione dell'infrastruttura cloud con. AWS CDK
- Segui le migliori pratiche di sicurezza in IAM quando utilizzi i flussi di lavoro Servizi AWS in GitHub Actions, tra cui:
  - Non memorizzate le credenziali nel codice del repository.
  - Assumi un ruolo IAM per ricevere credenziali temporanee e utilizza credenziali temporanee quando possibile.
  - Concedi il privilegio minimo al ruolo IAM utilizzato nei GitHub flussi di lavoro di Actions. Concedi solo le autorizzazioni necessarie per eseguire le azioni nei flussi di lavoro Actions. GitHub
  - Monitora l'attività del ruolo IAM utilizzato nei flussi di lavoro di GitHub Actions.
  - Ruota periodicamente tutte le credenziali a lungo termine che utilizzi.

Epiche

Configurare l'ambiente

Operazione	Description	Competenze richieste
Configura AWS le credenziali.	Per esportare le variabili che definiscono lo stack Account AWS e il Regione AWS luogo	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste

	in cui stai distribuendo, esegui i seguenti comandi:
	`export CDK_DEFAULT_ACCOUNT=<12-digit AWS account number>`
	`export CDK_DEFAULT_REGION=<AWS Region>`
	Le AWS credenziali per AWS CDK vengono fornite tramite variabili di ambiente.	

Clonare il repository.	Per clonare il repository sul computer locale, esegui il seguente comando: `git clone https://github.com/aws-samples/amazon-codeguru-suite-cdk-python.git`	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste

Installa il CDK Toolkit.	Per confermare che CDK Toolkit sia installato e per verificarne la versione, esegui il seguente comando:
	`cdk --version`
	Se la versione di CDK Toolkit è precedente alla 2.27.0, inserisci il seguente comando per aggiornarla alla versione 2.27.0:
	`npm install -g aws-cdk@2.27.0`
	Se CDK Toolkit non è installato, esegui il seguente comando per installarlo:
	`npm install -g aws-cdk@2.27.0 --force`
		AWS DevOps, DevOps ingegnere

Installare le dipendenze richieste.	Per installare le dipendenze richieste dal progetto, esegui il seguente comando:
	`python -m pip install --upgrade pip`
	`pip install -r requirements.txt`
		AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste

Avvia l'ambiente CDK.	Per avviare un ambiente AWS CDK, esegui i seguenti comandi:
	`npm install`
	`npm run cdk bootstrap "aws://${ACCOUNT_NUMBER}/${AWS_REGION}"`
	Dopo aver avviato correttamente l'ambiente, dovrebbe essere visualizzato il seguente output:
	`# Bootstrapping environment aws://{account}/{region}...`
	`# Environment aws://{account}/{region} bootstrapped`
		AWS DevOps, DevOps ingegnere

Implementa l'app CDK

Operazione	Description	Competenze richieste
Sintetizza l'app. AWS CDK	Per sintetizzare un' AWS CDK app, esegui il seguente comando: `cdk synth` Per ulteriori informazioni su questo comando, vedete cdk synthesize nella documentazione. AWS CDK	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
Distribuisci le risorse.	Per distribuire le risorse, esegui il comando seguente: `cdk deploy --require-approval never` **Note** Il `--require-approval never` flag indica che il CDK approverà ed eseguirà automaticamente tutte le modifiche. Ciò include le modifiche che il CDK normalmente contrassegnerebbe come richiedenti una revisione manuale (come le modifiche alle politiche IAM o la rimozione di risorse). Assicurati che il codice CDK e la CI/CD pipeline siano ben testati e sicuri prima di utilizzare il `--require-approval never` flag negli ambienti di produzione.	AWS DevOps, DevOps ingegnere

Crea GitHub segreti e token di accesso personale

Operazione	Description	Competenze richieste
Crea i segreti richiesti in GitHub.	Per consentire ai flussi di lavoro di GitHub Actions di accedere alle AWS risorse in modo sicuro senza esporre informazioni sensibili nel codice del repository, crea dei segreti. Per creare i segreti in GitHub for`ROLE_TO_ASSUME` , and `CodeGuruReviewArtifactBucketName` `AWS_ACCOUNT_ID` , segui le istruzioni in Creazione di segreti per un repository nella documentazione di Actions. GitHub Di seguito sono riportate ulteriori informazioni sulle variabili: • `AWS_ACCOUNT_ID` — L'Account AWS ID in cui viene eseguita la pipeline. • `CodeGuruReviewArtifactBucketName` — Il nome del bucket S3 in cui sono archiviati gli artefatti di CodeGuru Reviewer. Questo modello utilizza il nome del bucket. `codeguru-reviewer-build-artifacts-<A`	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	CCOUNT_ID>-<REGION> • AWS_REGION — Regione AWS Dove si trovano le risorse. • ROLE_TO_ASSUME — Il nome del ruolo IAM assunto dalla pipeline. Questo modello utilizza il nome del ruolo. githubActionsDeployRole	
Crea un token di accesso GitHub personale.	Per configurare un modo sicuro con cui autenticarsi e interagire con i flussi di lavoro GitHub Actions GitHub, procedi come segue: 1. Per creare un token di accesso GitHub personale con accesso in lettura e scrittura al tuo repository, segui le istruzioni in Gestione dei token di accesso personali nella documentazione. GitHub 2. Per salvare questo token come repository secret per GitHub Actions, segui le istruzioni in Creazione di segreti per un repository nella documentazione Actions. GitHub	AWS DevOps, DevOps ingegnere

Eliminazione

Operazione	Description	Competenze richieste
Eliminare le risorse.	Per pulire la tua app AWS CDK Python, esegui il seguente comando: `cdk destroy --all`	DevOps ingegnere

Risoluzione dei problemi

Problema	Soluzione
Visualizza il link ai risultati della dashboard.	Non è possibile stampare i risultati durante la CI/CD pipeline. Invece, questo modello utilizza GitHub Actions come metodo alternativo per gestire e visualizzare i risultati.

Risorse correlate

AWS resources

- AWS Cloud Development Kit
- CodeGuru Documentazione Amazon
- Amazon S3
- AWS Identity and Access Management
- Amazon Simple Queue Service
- Che cos'è AWS Lambda?

GitHub documentazione

- Configurazione di OpenID Connect in Amazon Web Services
- GitHub Azioni

- Riutilizzo dei flussi di lavoro
- Attivazione di un flusso di lavoro

Automatizza l'implementazione dei Catena di approvvigionamento di AWS data lake in una configurazione multi-repository utilizzando GitHub Actions, Artifactory e Terraform

Keshav Ganesh, Amazon Web Services

Riepilogo

Questo modello fornisce un approccio automatizzato per l'implementazione e la gestione Catena di approvvigionamento di AWS dei data lake utilizzando funzionalità di integrazione continua multirepository e distribuzione continua (). CI/CD) pipeline. It demonstrates two deployment methods: automated deployment using GitHub Actions workflows, or manual deployment using Terraform directly. Both approaches use Terraform for infrastructure as code (IaC), with the automated method adding GitHub Actions and JFrog Artifactory for enhanced CI/CD

La soluzione sfrutta Catena di approvvigionamento di AWS Amazon Simple Storage Service (Amazon S3) per stabilire l'infrastruttura del data lake, utilizzando entrambi i metodi di distribuzione per automatizzare la configurazione e la creazione di risorse. AWS Lambda Questa automazione elimina le fasi di configurazione manuali e garantisce implementazioni coerenti in tutti gli ambienti. Inoltre, Catena di approvvigionamento di AWS elimina la necessità di competenze approfondite in materia di estrazione, trasformazione e caricamento (ETL) e può fornire approfondimenti e analisi basati su Amazon Quick Sight.

Implementando questo modello, le organizzazioni possono ridurre i tempi di implementazione, mantenere l'infrastruttura come codice e gestire i data lake della catena di fornitura attraverso un processo automatizzato e controllato dalla versione. L'approccio multirepository fornisce un controllo granulare degli accessi e supporta l'implementazione indipendente di diversi componenti. I team possono scegliere il metodo di implementazione più adatto agli strumenti e ai processi esistenti.

Prerequisiti e limitazioni

Prerequisiti

Assicurati che sul tuo computer locale sia installato quanto segue:

- AWS Command Line Interface (AWS CLI) versione 2
- GitHub CLI
- Python v3.13

- Terraform v1.12 o versione successiva

Assicurati che siano presenti le seguenti condizioni prima della distribuzione:

- Un attivo Account AWS.
- Un cloud privato virtuale (VPC) con due sottoreti private Account AWS a tua scelta. Regione AWS
- Autorizzazioni sufficienti per il ruolo AWS Identity and Access Management (IAM) utilizzato per l'implementazione nei seguenti servizi:
  - Catena di approvvigionamento di AWS — Accesso completo preferito per l'implementazione dei relativi componenti, come set di dati e flussi di integrazione, oltre ad accedervi da. Console di gestione AWS
  - Amazon CloudWatch Logs: per creare e gestire gruppi di CloudWatch log.
  - Amazon Elastic Compute Cloud (Amazon EC2) — Per i gruppi di sicurezza Amazon EC2 e gli endpoint Amazon Virtual Private Cloud (Amazon VPC).
  - Amazon EventBridge : destinato all'uso da Catena di approvvigionamento di AWS.
  - IAM: per creare ruoli AWS Lambda di servizio.
  - AWS Key Management Service (AWS KMS) — Per accedere al bucket AWS KMS keys utilizzato per il bucket di artefatti Amazon S3 e il bucket di staging Amazon S3. Catena di approvvigionamento di AWS
  - AWS Lambda — Per creare le funzioni Lambda che distribuiscono i componenti. Catena di approvvigionamento di AWS
  - Amazon S3: per accedere al bucket di artefatti Amazon S3, al bucket di registrazione degli accessi al server e al bucket di staging. Catena di approvvigionamento di AWS Se utilizzi la distribuzione manuale, sono necessarie anche le autorizzazioni per il bucket di artefatti Amazon S3 Terraform.
  - Amazon VPC: per creare e gestire un VPC.

Se preferisci utilizzare i flussi di lavoro GitHub Actions per la distribuzione, procedi come segue:

- Configura OpenID Connect (OIDC) per il ruolo IAM con le autorizzazioni menzionate in precedenza.
- Crea un ruolo IAM con autorizzazioni simili per accedere a. Console di gestione AWS Per ulteriori informazioni, consulta Creare un ruolo per concedere le autorizzazioni a un utente IAM nella documentazione IAM.

Se preferisci eseguire una distribuzione manuale, procedi come segue:

- Crea un utente IAM che assuma il ruolo IAM con le autorizzazioni menzionate in precedenza. Per ulteriori informazioni, consulta Creare un ruolo per concedere le autorizzazioni a un utente IAM nella documentazione IAM.
- Assumi il ruolo nel tuo terminale locale.

Se preferisci utilizzare i flussi di lavoro di GitHub Actions per la distribuzione, configura quanto segue:

- Un account JFrog Artifactory per ottenere il nome host, il nome utente di accesso e il token di accesso di accesso.
- Una chiave di JFrog progetto e un archivio per archiviare gli artefatti.

Limitazioni

- L' Catena di approvvigionamento di AWS istanza non supporta tecniche complesse di trasformazione dei dati.
- Catena di approvvigionamento di AWS è più adatto per i domini della catena di fornitura perché fornisce analisi e approfondimenti integrati. Per qualsiasi altro dominio, Catena di approvvigionamento di AWS può essere utilizzato come archivio dati come parte dell'architettura del data lake.
- Potrebbe essere necessario migliorare le funzioni Lambda utilizzate in questa soluzione per gestire i tentativi di API e la gestione della memoria in un'implementazione su scala di produzione.
- Alcune Servizi AWS non sono disponibili in tutte. Regioni AWS Per informazioni sulla disponibilità per regione, consulta AWS Servizi per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Architecture

Puoi implementare questa soluzione utilizzando i flussi di lavoro GitHub Actions automatizzati o manualmente utilizzando Terraform.

Distribuzione automatizzata con Actions GitHub

Il diagramma seguente mostra l'opzione di distribuzione automatizzata che utilizza i flussi di lavoro GitHub Actions. JFrog Artifactory viene utilizzato per la gestione degli artefatti. Memorizza le informazioni sulle risorse e gli output per l'uso in una distribuzione multirepository.

Distribuzione manuale con Terraform

Il diagramma seguente mostra l'opzione di distribuzione manuale tramite Terraform. Invece di JFrog Artifactory, Amazon S3 viene utilizzato per la gestione degli artefatti.

Workflow di implementazione

I diagrammi mostrano il seguente flusso di lavoro:

1. Distribuisci l'infrastruttura e i database dei set di dati dei Catena di approvvigionamento di AWS servizi utilizzando uno dei seguenti metodi di distribuzione:
   - Distribuzione automatizzata: utilizza i flussi di lavoro di GitHub Actions per orchestrare tutte le fasi di distribuzione e utilizza JFrog Artifactory per la gestione degli artefatti.
   - Distribuzione manuale: esegue i comandi Terraform direttamente per ogni fase di distribuzione e utilizza Amazon S3 per la gestione degli artefatti.
2. Crea le AWS risorse di supporto necessarie per il funzionamento del servizio: Catena di approvvigionamento di AWS
   - Endpoint e gruppi di sicurezza Amazon VPC
   - AWS KMS keys
   - CloudWatch Registra i gruppi di log
3. Crea e distribuisci le seguenti risorse di infrastruttura:
   - Funzioni Lambda che gestiscono (creano, aggiornano ed eliminano) l'istanza del Catena di approvvigionamento di AWS servizio, i namespace e i set di dati.
   - Catena di approvvigionamento di AWS gestione temporanea del bucket Amazon S3 per l'inserimento di dati
4. Implementa la funzione Lambda che gestisce i flussi di integrazione tra lo staging bucket e i set di dati. Catena di approvvigionamento di AWS Una volta completata l'implementazione, le fasi rimanenti del flusso di lavoro gestiscono l'inserimento e l'analisi dei dati.
5. Configura l'inserimento dei dati di origine nel bucket di Catena di approvvigionamento di AWS staging Amazon S3.
6. Dopo aver aggiunto i dati al bucket Catena di approvvigionamento di AWS di staging Amazon S3, il servizio attiva automaticamente il flusso di integrazione nei set di dati. Catena di approvvigionamento di AWS

7. Catena di approvvigionamento di AWS si integra con Quick Sight Analytics per produrre dashboard basate sui dati acquisiti.

Tools (Strumenti)

Servizi AWS

- Amazon CloudWatch Logs ti aiuta a centralizzare i log di tutti i tuoi sistemi e applicazioni, Servizi AWS così puoi monitorarli e archiviarli in modo sicuro.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire Servizi AWS tramite comandi nella shell della riga di comando.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel Cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altro modo. Account AWS
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS IAM Identity Centerti aiuta a gestire centralmente l'accesso Single Sign-On (SSO) a tutte le tue applicazioni e a quelle sul cloud. Account AWS
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Q in Catena di approvvigionamento di AWS è un assistente AI generativo interattivo che ti aiuta a gestire la catena di approvvigionamento in modo più efficiente analizzando i dati nel tuo Catena di approvvigionamento di AWS data lake.
- Amazon Quick Sight è un servizio di business intelligence (BI) su scala cloud che ti aiuta a visualizzare, analizzare e generare report sui dati in un'unica dashboard.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

- Catena di approvvigionamento di AWSè un'applicazione gestita basata sul cloud che può essere utilizzata come archivio dati nelle organizzazioni per i domini della catena di fornitura, che può essere utilizzata per generare approfondimenti ed eseguire analisi sui dati acquisiti.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS. Un endpoint Amazon VPC è un dispositivo virtuale che ti aiuta a connettere privatamente il tuo VPC a Supported Servizi AWS senza richiedere un gateway Internet, un dispositivo NAT, una connessione VPN o una connessione. AWS Direct Connect

Altri strumenti

- GitHub Actions è una piattaforma di integrazione e distribuzione continua (CI/CD) strettamente integrata con i repository. GitHub Puoi utilizzare GitHub Actions per automatizzare la pipeline di compilazione, test e distribuzione.
- HashiCorp Terraform è uno strumento di infrastruttura come codice (IaC) che ti aiuta a creare e gestire risorse cloud e locali.
- JFrog Artifactory fornisce l' end-to-endautomazione e la gestione di file binari e artefatti attraverso il processo di distribuzione delle applicazioni.
- Python è un linguaggio di programmazione per computer generico. Questo modello utilizza Python con cui il codice della AWS funzione può interagire Catena di approvvigionamento di AWS

.

Best practice

- Mantieni la massima sicurezza possibile durante l'implementazione di questo modello. Come indicato in Prerequisiti, assicurati che un cloud privato virtuale (VPC) con due sottoreti private sia presente nel Account AWS Regione AWS tuo computer di tua scelta.
- Utilizza le chiavi gestite AWS KMS dal cliente laddove possibile e concedi loro autorizzazioni di accesso limitate.
- Per configurare i ruoli IAM con il minimo accesso richiesto per l'acquisizione dei dati per questo pattern, consulta Secure Data Ingestion from Source Systems to Amazon S3 nel repository di questo pattern.

Epiche

(Entrambe le opzioni) Configura una workstation locale

Operazione	Description	Competenze richieste

Clonare il repository.	Per clonare il repository di questo pattern, esegui il seguente comando nella tua workstation locale:
	`git clone https://github.com/aws-samples/sample-automate-aws-supply-chain-deployment.git`
	`cd ASC-Deployment`
		AWS DevOps

(Opzione automatizzata) Verifica i prerequisiti per la distribuzione.	Assicurati che i prerequisiti per la distribuzione automatizzata siano completi.	Proprietario dell'app

(Opzione manuale) Preparati per la distribuzione dei Catena di approvvigionamento di AWS set di dati.	Per accedere alla `terraform-deployment` directory di`ASC-Datasets` , esegui il seguente comando:
	`cd ASC-Datasets/terraform-deployment`
	Per assumere il ruolo ARN creato nei Prerequisiti, esegui il comando seguente:
	`aws sts assume-role --role-arn <enter AWS user role ARN> --role-session-name <your-session-name>`
		AWS DevOps

Operazione	Description	Competenze richieste

	Per configurare ed esportare le variabili di ambiente, esegui i seguenti comandi:
	
	`# Export Environment variables`
	`export REGION=<Enter deployment region>`
	`export REPO_NAME=<Enter Current ASC Datasets dir name>`
	`export PROJECT_NAME="asc-deployment-poc"`
	`export ACCOUNT_ID=<Enter deployment Account ID>`
	`export ENVIRONMENT="dev"`
	`export LAMBDA_LAYER_TEMP_DIR_TERRAFORM="layerOutput"`
	`export LAMBDA_FUNCTION_TEMP_DIR_TERRAFORM="lambdaOutput"`
	`export AWS_USER_ROLE=<Enter user role ARN for AWS Console access and deployment>`
	`export S3_TERRAFORM_ARTIFACTS_BUCKET_NAME="$PROJECT_NAME-$ACCOUNT_ID-$REGION-terraform-artifacts-$ENVIRONMENT"`	

Operazione	Description	Competenze richieste

(Opzione manuale) Preparati a gestire i flussi di Catena di approvvigionamento di AWS integrazione durante la distribuzione.	Per accedere alla `terraform-deployment` directory di`ASC-Integration-Flows` , esegui il seguente comando:
	`cd ASC-Integration-Flows/terraform-deployment`
	Per assumere il ruolo ARN creato in precedenza, esegui il seguente comando:
	`aws sts assume-role --role-arn <enter AWS user role ARN> --role-session-name <your-session-name>`
	Per configurare ed esportare le variabili di ambiente, esegui i seguenti comandi:
	`# Export Environment variables`
	`export REGION=<Enter deployment region>`
	`export REPO_NAME=<Enter Current ASC Integration Flows dir name>`
	`export ASC_DATASET_VARS_REPO=<Enter Current ASC Datasets dir name>  #Must be the same directory name`
		Proprietario dell'app

Operazione	Description	Competenze richieste
	used for ASC Datasets deployment export PROJECT_NAME="asc-deployment-poc" export ACCOUNT_ID=<Enter deployment Account ID> export ENVIRONMENT="dev" export LAMBDA_LAYER_TEMP_DIR_TERRAFORM="layerOutput" export LAMBDA_FUNCTION_TEMP_DIR_TERRAFORM="lambdaOutput" export S3_TERRAFORM_ARTIFACTS_BUCKET_NAME="$PROJECT_NAME-$ACCOUNT_ID-$REGION-terraform-artifacts-$ENVIRONMENT"	

(Opzione automatizzata) Distribuisci Catena di approvvigionamento di AWS set di dati utilizzando GitHub i flussi di lavoro Actions

Operazione	Description	Competenze richieste
Copia la `ASC-Datasets` cartella.	Per copiare la `ASC-Datasets` directory in una nuova posizione, procedi nel seguente modo: 1. Per accedere alla `ASC-Datasets` directory	AWS DevOps

Operazione	Description	Competenze richieste

	, esegui il comando seguente:
	`cd ASC-Datasets`
	2. Per copiare la `ASC-Datasets` directory in una nuova posizione, esegui i seguenti comandi:
	`cp -r ASC-Datasets ../ASC-Datasets-standalone`
	`cd ../ASC-Datasets-standalone`	

Configura la `ASC-Datasets` directory.	Per configurarlo `ASC-Datasets` come repository autonomo nell'organizzazione, esegui i seguenti comandi:
	`git init`
	`git add .`
	`git commit -m "Initial commit: ASC-Datasets standalone repository"`
	`git remote add origin <INSERT_ASC_DATASETS_GITHUB_URL>`
	`git branch -M dev`
		AWS DevOps

Operazione	Description	Competenze richieste

Configura il nome del ramo nel file di workflow .github.	Imposta il nome del ramo nel file del flusso di lavoro di distribuzione come mostrato nell'esempio seguente:
	`on:`
	`workflow_dispatch:`
	`push:`
	`branches:`
	`- dev`
	`#Change to any other branch preferred for deployment`
		Proprietario dell'app

Configura GitHub gli ambienti e configura i valori dell'ambiente.	Per configurare GitHub gli ambienti nella propria GitHub organizzazione, utilizzare le istruzioni in Configurazione GitHub degli ambienti nell'archivio di questo modello. Per configurare i valori di ambiente nei file del flusso di lavoro, utilizzate le istruzioni in Configurazione dei valori di ambiente nei file di flusso di lavoro nel repository di questo pattern.	Proprietario dell'app

Operazione	Description	Competenze richieste
Attiva il flusso di lavoro.	Per inviare le modifiche all' GitHub organizzazione e attivare il flusso di lavoro di distribuzione, esegui il comando seguente: `git push -u origin dev`	AWS DevOps

(Opzione automatizzata) Distribuisci flussi di Catena di approvvigionamento di AWS integrazione utilizzando i flussi di lavoro GitHub Actions

Operazione	Description	Competenze richieste

Copia la `ASC-Integration-Flows` cartella.	Per copiare la `ASC-Integration-Flows` directory in una nuova posizione, procedi nel seguente modo:
	1. Per accedere alla `ASC-Integration-Flows` directory, esegui il comando seguente:
	`cd ASC-Integration-Flows`
	2. Per copiare la `ASC-Integration-Flows` directory in una nuova posizione, esegui i seguenti comandi:
	`cp -r ASC-Integration-Flows ../ASC-`
		AWS DevOps

Operazione	Description	Competenze richieste
	`Integration-Flows-st andalone cd ../ASC-In tegration-Flows-st andalone`	
Configura la `ASC-Integ ration-Flows` directory.	Per configurare la `ASC-Integ ration-Flows` directory come archivio autonomo nell'organizzazione, esegui i seguenti comandi: `git init git add . git commit -m "Initial commit: ASC-Integ ration-Flows standalon e repository" git remote add origin <INSERT_ASC_Integr ation_Flows_GITHUB _URL> git branch -M dev`	AWS DevOps
Configura il nome del ramo nel file di workflow .github.	Imposta il nome del ramo nel file del flusso di lavoro di distribuzione come mostrato nell'esempio seguente: `on: workflow_ dispatch: push: branches: - dev #Change to any other branch preferred for deployment`	Proprietario dell'app

Operazione	Description	Competenze richieste
Configura GitHub gli ambienti e configura i valori dell'ambiente.	Per configurare GitHub gli ambienti nella propria GitHub organizzazione, utilizzare le istruzioni in Configurazione GitHub degli ambienti nell'archivio di questo modello. Per configurare i valori di ambiente nei file del flusso di lavoro, utilizzate le istruzioni in Configurazione dei valori di ambiente nei file di flusso di lavoro nel repository di questo pattern.	Proprietario dell'app
Attiva il flusso di lavoro.	Per inviare le modifiche all' GitHub organizzazione e attivare il flusso di lavoro di distribuzione, esegui il comando seguente: `git push -u origin dev`	AWS DevOps

(Opzione manuale) Distribuisci Catena di approvvigionamento di AWS set di dati utilizzando Terraform

Operazione	Description	Competenze richieste
Passa alla directory `terraform-deployment` .	Per accedere alla `terraform-deployment` directory di`ASC-Datasets` , esegui il seguente comando:	AWS DevOps

Operazione	Description	Competenze richieste
	`cd ASC-Datasets/terraform-deployment`	

Configura il bucket Amazon S3 dello stato Terraform.	Per configurare il bucket Amazon S3 dello stato Terraform, usa lo script seguente:
	`# Setup terraform bucket`
	`chmod +x ../scripts/setup-terraform.sh`
	`../scripts/setup-terraform.sh`
		AWS DevOps

Configura il bucket Terraform Artifacts Amazon S3.	Per configurare il bucket Terraform Artifacts Amazon S3, usa lo script seguente:
	`# Setup terraform artifacts bucket`
	`chmod +x ../scripts/setup-terraform-artifacts-bucket.sh`
	`../scripts/setup-terraform-artifacts-bucket.sh`
		AWS DevOps

Operazione	Description	Competenze richieste

Configura il backend Terraform e la configurazione dei provider.	Per configurare il backend Terraform e la configura zione dei provider, utilizza il seguente script:
	`# Setup terraform backend and providers config if they don't exist`
	`chmod +x ../scripts/generate-terraform-config.sh`
	`../scripts/generate-terraform-config.sh`
		AWS DevOps

Operazione	Description	Competenze richieste

Genera un piano di distribuzione.	Per generare un piano di distribuzione, esegui i seguenti comandi:
	`# Run terraform init and validate`
	`terraform init`
	`terraform validate`
	
	`# Run terraform plan`
	`terraform plan \`
	`-var-file="tfInputs/$ENVIRONMENT.tfvars" \`
	`-var="project_name=$PROJECT_NAME" \`
	`-var="environment=$ENVIRONMENT" \`
	`-var="user_role=$AWS_USER_ROLE" \`
	`-var="lambda_temp_dir=$LAMBDA_FUNCTION_TEMP_DIR_TERRAFORM" \`
	`-var="layer_temp_dir=$LAMBDA_LAYER_TEMP_DIR_TERRAFORM" \`
	`-parallelism=40 \`
	`-out='tfplan.out'`
		AWS DevOps

Implementa le configurazioni.	Per distribuire le configurazioni, esegui il comando seguente:
	`# Run terraform apply`
	`terraform apply`
	`tfplan.out`
		AWS DevOps

Operazione	Description	Competenze richieste

Aggiorna altre configurazioni e archivia gli output.	Per aggiornare le politiche AWS KMS chiave e archiviare gli output delle configurazioni applicate nel bucket Terraform Artifacts Amazon S3, esegui i seguenti comandi:
	`# Update AWS Supply Chain KMS Key policy with the service's requirements`
	`chmod +x ../scripts/update-asc-kms-policy.sh`
	`../scripts/update-asc-kms-policy.sh`
	`# Update AWS KMS Keys' policy with IAM roles`
	`chmod +x ../scripts/update-kms-policy.sh`
	`../scripts/update-kms-policy.sh`
	`# Create terraform outputs file to be used as input variables`
	`terraform output -json > raw_output.json`
	`jq -r 'to_entries | map( if .value.type == "string" then "\(.key) = \"\(.value.value)\"" else`
		AWS DevOps

Operazione	Description	Competenze richieste

	`"\(.key) = \(.value.value | tojson)" end ) | .[]' raw_output.json > $REPO_NAME-outputs.tfvars`
	
	`# Upload reformed outputs file to Amazon S3 terraform artifacts bucket (For retrieval from other repositories) aws s3 cp $REPO_NAME-outputs.tfvars s3://$S3_TERRAFORM_ARTIFACTS_BUCKET_NAME/$REPO_NAME-outputs.tfvars rm -f raw_output.json rm -f $REPO_NAME-outputs.tfvars`	

(Opzione manuale) Distribuisci i flussi Catena di approvvigionamento di AWS di integrazione dei servizi utilizzando Terraform

Operazione	Description	Competenze richieste
Passa alla directory `terraform-deployment` .	Per accedere alla `terraform-deployment` directory di`ASC-Integration-Flows` , esegui il seguente comando:	AWS DevOps

Operazione	Description	Competenze richieste
	`cd ASC-Integration-Flows/terraform-deployment`	

Configura il backend Terraform e la configurazione dei provider.	Per configurare il backend Terraform e le configurazioni del provider, utilizza il seguente script:
	`# Setup terraform backend and providers config if they don't exist`
	`chmod +x ../scripts/generate-terraform-config.sh`
	`../scripts/generate-terraform-config.sh`
		AWS DevOps

Operazione	Description	Competenze richieste

Genera un piano di distribuzione.	Per generare un piano di distribuzione, esegui i comandi seguenti. Questi comandi inizializzano l'ambiente Terraform, uniscono le variabili di configurazione `ASC-Datasets` con le configurazioni Terraform esistenti e generano un piano di implementazione.
	`# Run terraform init and validate`
	`terraform init`
	`terraform validate`
	`# Download and merge ASC DATASET tfvars`
	`chmod +x ../scripts/download-vars-through-s3.sh`
	`../scripts/download-vars-through-s3.sh $ASC_DATASET_VARS_REPO`
	`# Run terraform plan`
	`terraform plan \`
	`-var-file="tfInputs/$ENVIRONMENT.tfvars" \`
	`-var="project_name=$PROJECT_NAME" \`
	`-var="environment=$ENVIRONMENT" \`
	`-var="lambda_temp_dir=$LAMBDA_FUNCTION_TEMP_DIR_TERRAFORM" \`
		AWS DevOps

Operazione	Description	Competenze richieste
	`-var="layer_temp_dir=$LAMBDA_LAYER_TEMP_DIR_TERRAFORM" \ -parallelism=40 \ -out='tfplan.out'`	
Implementa le configurazioni.	Per distribuire le configurazioni, esegui il comando seguente: `# Run terraform apply terraform apply tfplan.out`	AWS DevOps

Operazione	Description	Competenze richieste

Aggiorna altre configurazioni.	Per aggiornare le politiche AWS KMS chiave e archiviare gli output delle configurazioni applicate nel bucket Terraform Artifacts Amazon S3, esegui i seguenti comandi:
	`# Update AWS KMS Keys' policy with IAM roles`
	`chmod +x ../scripts/update-kms-policy-through-s3.sh`
	`../scripts/update-kms-policy-through-s3.sh $ASC_DATA SET_VARS_REPO`
	`# Create terraform outputs file to be used as input variables`
	`terraform output -json > raw_output.json`
	`jq -r 'to_entries | map( if .value.type == "string" then "\(.key) = \"\(.value.value)\"" else "\(.key) = \(.value.value | tojson)" end ) | .[]' raw_output.json > $REPO_NAME-outputs.tfvars`
		AWS DevOps

Operazione	Description	Competenze richieste

	`# Upload reformed outputs file to Amazon S3 terraform artifacts bucket (For retrieval from other repositories)`
	`aws s3 cp $REPO_NAME-outputs.tfvars s3://$S3_TERRAFORM_ARTIFACTS_BUCKET_NAME/$REPO_NAME-outputs.tfvars`
	`rm -f raw_output.json`
	`rm -f $REPO_NAME-outputs.tfvars`	

(Entrambe le opzioni) Inserisci dati

Operazione	Description	Competenze richieste
Carica file CSV di esempio.	Per caricare file CSV di esempio per i set di dati, procedi nel seguente modo: 1. Crea file CSV di esempio con vari dati per i set di dati Calendar e Outbound Order Line creati durante la distribuzione. 2. Recupera l'ID dell' Catena di approvvigionamento di AWS istanza `asc_instance_id` dalla directory degli output di terraform. 3. Nota il nome del bucket Amazon S3 per il quale è stato creato Catena di	Ingegnere dei dati

Operazione	Description	Competenze richieste

	approvvigionamento di AWS durante la distribuzione: `aws-supply-chain-data-<Instance_ID>`
	4. Per caricare i file utilizzando il AWS CLI, esegui i seguenti comandi:
	`# Upload Calendar CSV file`
	`aws s3 cp calendar_sample.csv s3://aws-supply-chain-data-<Instance_ID>/calendar-data/`
	`# Upload Outbound Order Line CSV file`
	`aws s3 cp outbound_order_line_sample.csv s3://aws-supply-chain-data-<Instance_ID>/outbound-order-line-data/`	

(Entrambe le opzioni) Configura Catena di approvvigionamento di AWS l'accesso

Operazione	Description	Competenze richieste
Configura Catena di approvvigionamento di AWS l'accesso.	Per configurare Catena di approvvigionamento di AWS l'accesso da Console di gestione AWS, procedi nel seguente modo:	Proprietario dell'app

Operazione	Description	Competenze richieste
	1. Accedi a Console di gestione AWS e cerca il Catena di approvvigionamento di AWS servizio. 2. Vai all'istanza`asc-deployment-poc-dev-asc-instance` . 3. Questo modello utilizza IAM Identity Center per gestire l'accesso degli utenti all' Catena di approvvigionamento di AWS istanza. Per garantire l'accesso completo a questa soluzione, accedi come amministratore dell'applicazione.	

(Opzione automatizzata) Pulisci tutte le risorse utilizzando i flussi di lavoro GitHub Actions

Operazione	Description	Competenze richieste
Attiva il flusso di lavoro Destroy per le risorse dei flussi di integrazione.	Attiva il flusso di lavoro di distruzione `ASC-Integration-Flows` dal tuo ramo di distribuzione GitHub all'interno dell'organizzazione.	AWS DevOps
Attiva il flusso di lavoro di distruzione delle risorse dei set di dati.	Attiva il flusso di lavoro di distruzione `ASC-Datasets` dal tuo ramo di distribuzione all'interno GitHub dell'organizzazione.	AWS DevOps

(Opzione manuale) Pulisci le risorse dai flussi di Catena di approvvigionamento di AWS integrazione utilizzando Terraform

Operazione	Description	Competenze richieste
Passa alla directory `terraform-deployment` .	Per accedere alla `terraform-deployment` directory di`ASC-Integration-Flows` , esegui il seguente comando: `cd ASC-Integration-Flows/terraform-deployment`	AWS DevOps

Configura il backend Terraform e la configurazione dei provider.	Per configurare il backend Terraform e la configurazione dei provider, utilizza il seguente script:
	`# Setup terraform backend and providers config if they don't exist`
	`chmod +x ../scripts/generate-terraform-config.sh`
	`../scripts/generate-terraform-config.sh`
		AWS DevOps

Genera un piano di distruzione dell'infrastruttura.	Per prepararti alla distruzione controllata dell' AWS infrastruttura generando un piano di smontaggio dettagliato, esegui i seguenti comandi. Il processo inizializza Terraform , incorpora le configurazioni Catena di approvvigionamento	AWS DevOps

Operazione	Description	Competenze richieste

	di AWS dei set di dati e crea un piano di distruzione che è possibile esaminare prima dell'esecuzione.
	
	`# Run terraform init and validate`
	`terraform init`
	`terraform validate`
	
	`# Download and merge ASC DATASET tfvars`
	`chmod +x ../scripts/download-vars-through-s3.sh`
	`../scripts/download-vars-through-s3.sh $ASC_DATASET_VARS_REPO`
	
	`# Run terraform plan`
	`terraform plan -destroy \`
	`-var-file="tfInputs/$ENVIRONMENT.tfvars" \`
	`-var="project_name=$PROJECT_NAME" \`
	`-var="environment=$ENVIRONMENT" \`
	`-var="lambda_temp_dir=$LAMBDA_FUNCTION_TEMP_DIR_TERRAFORM" \`
	`-var="layer_temp_dir=$LAMBDA_LAYER_TEMP_DIR_TERRAFORM" \`
	`-parallelism=40 \`
	`-out='tfplan.out'`	

Operazione	Description	Competenze richieste

Esegui un piano di distruzione dell'infrastruttura.	Per eseguire la distruzione pianificata dell'infrastruttura, esegui il seguente comando:
	`# Run terraform apply`
	`terraform apply`
	`tfplan.out`
		AWS DevOps

Rimuovi gli output Terraform dal bucket Amazon S3.	Per rimuovere il file di output che è stato caricato durante la distribuzione di`ASC-Integration-Flows`, esegui il seguente comando:
	`# Delete the outputs file`
	`aws s3 rm s3://$S3_TERRAFORM_ARTIFACTS_BUCKET_NAME/$REPO_NAME-outputs.tfvars`
		AWS DevOps

(Opzione manuale) Pulisci le risorse dei set di dati di Catena di approvvigionamento di AWS servizio utilizzando Terraform

Operazione	Description	Competenze richieste
Passa alla directory `terraform-deployment`.	Per accedere alla `terraform-deployment` directory di`ASC-Datasets`, esegui il seguente comando: `cd ASC-Datasets/terraform-deployment`	AWS DevOps

Operazione	Description	Competenze richieste

Configura il backend Terraform e la configurazione dei provider.	Per configurare il backend Terraform e la configura zione dei provider, utilizza il seguente script:
	`# Setup terraform backend and providers config if they don't exist`
	`chmod +x ../scripts/generate-terraform-config.sh`
	`../scripts/generate-terraform-config.sh`
		AWS DevOps

Operazione	Description	Competenze richieste

Genera un piano di distruzione dell'infrastruttura.	Per creare un piano per la distruzione delle risorse del Catena di approvvigionamento di AWS set di dati, esegui i seguenti comandi:
	`# Run terraform init and validate`
	`terraform init`
	`terraform validate`
	
	`# Run terraform plan`
	`terraform plan -destroy \`
	`-var-file="tfInputs/$ENVIRONMENT.tfvars" \`
	`-var="project_name=$PROJECT_NAME" \`
	`-var="environment=$ENVIRONMENT" \`
	`-var="user_role=$AWS_USER_ROLE" \`
	`-var="lambda_temp_dir=$LAMBDA_FUNCTION_TEMP_DIR_TERRAFORM" \`
	`-var="layer_temp_dir=$LAMBDA_LAYER_TEMP_DIR_TERRAFORM" \`
	`-parallelism=40 \`
	`-out='tfplan.out'`
		AWS DevOps

Operazione	Description	Competenze richieste

Bucket Amazon S3 vuoti.	Per svuotare tutti i bucket Amazon S3 (tranne il bucket di registrazione degli accessi al server, che è configurato per `force_destroy`), usa lo script seguente:
	`# Delete S3 buckets excluding server access logging bucket`
	`chmod +x ../scripts/empty-s3-buckets.sh`
	`../scripts/empty-s3-buckets.sh tfplan.out`
		AWS DevOps

Esegui un piano di distruzione dell'infrastruttura.	Per eseguire la distruzione pianificata dell'infrastruttura del Catena di approvvigionamento di AWS set di dati utilizzando il piano generato, esegui il comando seguente:
	`# Run terraform apply`
	`terraform apply tfplan.out`
		AWS DevOps

Operazione	Description	Competenze richieste

Rimuovi gli output Terraform dal bucket di artefatti Terraform di Amazon S3.	Per completare il processo di pulizia, rimuovi il file di output che è stato caricato durante la distribuzione di eseguendo il comando seguente: `ASC-Datasets`
	`# Delete the outputs file`
	`aws s3 rm s3://$S3_TERRAFORM_ARTIFACTS_BUCKET_NAME/$REPO_NAME-outputs.tfvars`
		AWS DevOps

Risoluzione dei problemi

Problema	Soluzione
Un Catena di approvvigionamento di AWS set di dati o un flusso di integrazione non è stato distribuito correttamente a causa di errori Catena di approvvigionamento di AWS interni o autorizzazioni IAM insufficienti per il ruolo di servizio.	Innanzitutto, pulisci tutte le risorse. Quindi, ridistribuisci le risorse del Catena di approvvigionamento di AWSset di dati e quindi ridistribuisci le risorse del flusso di integrazione. Catena di approvvigionamento di AWS
Il flusso di Catena di approvvigionamento di AWS integrazione non recupera i nuovi file di dati caricati per i set di dati. Catena di approvvigionamento di AWS	1. Verifica che il prefisso della configurazione del flusso di Catena di approvvigionamento di AWS integrazione corrisponda al prefisso utilizzato durante il caricamento dei file di dati di esempio. 2. Se le risorse per Catena di approvvigionamento di AWS i set di dati sono state ricreate, gli Amazon Resource Names (ARNs) associati cambiano internamente.

Problema	Soluzione
	Pertanto, ridistribuisci le risorse per le risorse dei flussi di integrazione dei Catena di approvvigionamento di AWS servizi.

Risorse correlate

AWS documentazione

- Catena di approvvigionamento di AWS

Altre risorse

- Comprensione dei flussi di lavoro di GitHub Actions (GitHub documentazione)

Informazioni aggiuntive

Questa soluzione può essere replicata per più set di dati e può essere interrogata per ulteriori analisi, tramite dashboard predefiniti forniti o integrazione Catena di approvvigionamento di AWS personalizzata con Amazon Quick Sight. Inoltre, puoi utilizzare Amazon Q per porre domande relative alla tua Catena di approvvigionamento di AWS istanza.

Analizza i dati con Catena di approvvigionamento di AWS Analytics

Per istruzioni su come configurare Catena di approvvigionamento di AWS Analytics, consulta Impostazione di Catena di approvvigionamento di AWS Analytics nella Catena di approvvigionamento di AWS documentazione.

Questo modello ha dimostrato la creazione di set di dati Calendar e Outbound_Order_Line. Per creare un'analisi che utilizzi questi set di dati, procedi nel seguente modo:

1. Per analizzare i set di dati, utilizza la dashboard di Seasonality Analysis. Per aggiungere la dashboard, segui i passaggi in Dashboard predefiniti nella documentazione. Catena di approvvigionamento di AWS
2. Scegli la dashboard per visualizzarne l'analisi basata su file CSV di esempio per i dati del calendario e i dati della linea degli ordini in uscita.

La dashboard fornisce informazioni sulla domanda nel corso degli anni sulla base dei dati acquisiti per i set di dati. È possibile specificare ulteriormente ProductID, CustomerId, years e altri parametri per l'analisi.

Usa Amazon Q per porre domande relative alla tua Catena di approvvigionamento di AWS istanza

Amazon Q in Catena di approvvigionamento di AWS è un assistente AI generativo interattivo che ti aiuta a gestire la catena di approvvigionamento in modo più efficiente. Amazon Q può fare quanto segue:

- Analizza i dati nel tuo Catena di approvvigionamento di AWS data lake.
- Fornisci informazioni operative e finanziarie.
- Rispondi alle tue domande immediate sulla catena di fornitura.

Per ulteriori informazioni sull'uso di Amazon Q, consulta la sezione Attivazione di Amazon Q in Catena di approvvigionamento di AWS e Uso di Amazon Q Catena di approvvigionamento di AWS nella Catena di approvvigionamento di AWS documentazione.

Automatizza la valutazione delle risorse AWS

Naveen Suthar, Arun Bagal, Manish Garg e Sandeep Gawande, Amazon Web Services

Riepilogo

Questo modello descrive un approccio automatizzato per la configurazione delle funzionalità di valutazione delle risorse utilizzando l'AWS Cloud Development Kit (AWS CDK). Utilizzando questo modello, i team operativi raccolgono i dettagli di controllo delle risorse in modo automatizzato e visualizzano i dettagli di tutte le risorse distribuite in un account AWS su un'unica dashboard. Ciò è utile nei seguenti casi d'uso:

- Identificazione degli strumenti Infrastructure as Code (IaC) e isolamento delle risorse create da diverse soluzioni IaC come HashiCorp Terraform, AWS CloudFormation, AWS CDK e AWS Command Line Interface (AWS CLI)
- Recupero di informazioni relative al controllo delle risorse

Questa soluzione aiuterà anche il team dirigenziale a ottenere informazioni sulle risorse e le attività in un account AWS da un'unica dashboard.

Nota: Amazon Quick Sight è un servizio a pagamento. Prima di eseguirlo per analizzare i dati e creare una dashboard, consulta i prezzi di Amazon Quick Sight.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Ruoli e autorizzazioni di AWS Identity and Access Management (IAM) con accesso alle risorse di provisioning
- Un account Amazon Quick creato con accesso ad Amazon Simple Storage Service (Amazon S3) e Amazon Athena
- AWS CDK versione 2.55.1 o successiva installata
- Python versione 3.9 o successiva installata

Limitazioni

- Questa soluzione viene distribuita su un singolo account AWS.
- La soluzione non terrà traccia degli eventi accaduti prima della sua implementazione a meno che AWS non CloudTrail fosse già stato configurato e archiviato i dati in un bucket S3.

Versioni del prodotto

- AWS CDK versione 2.55.1 o successiva
- Python versione 3.9 o successiva

Architecture

Stack tecnologico Target

- Amazon Athena
- AWS CloudTrail
- AWS Glue
- AWS Lambda
- Amazon Quick Sight
- Simple Storage Service (Amazon S3)

Architettura Target

Il codice CDK di AWS distribuirà tutte le risorse necessarie per configurare le funzionalità di valutazione delle risorse in un account AWS. Il diagramma seguente mostra il processo di invio dei CloudTrail log ad AWS Glue, Amazon Athena e Quick Sight.

1. CloudTrail invia i log a un bucket S3 per l'archiviazione.
2. Una notifica di evento richiama una funzione Lambda che elabora i log e genera dati filtrati.
3. I dati filtrati vengono archiviati in un altro bucket S3.
4. Un crawler AWS Glue è configurato sui dati filtrati presenti nel bucket S3 per creare uno schema nella tabella AWS Glue Data Catalog.
5. I dati filtrati sono pronti per essere interrogati da Amazon Athena.
6. I dati interrogati sono accessibili da Quick Sight per la visualizzazione.

Automazione e scalabilità

- Questa soluzione può essere scalata da un account AWS a più account AWS se esiste un percorso a livello di organizzazione in CloudTrail AWS Organizations. Implementandola CloudTrail a livello organizzativo, puoi utilizzare questa soluzione anche per recuperare i dettagli di controllo delle risorse per tutte le risorse richieste.
- Questo modello utilizza risorse serverless AWS per distribuire la soluzione.

Tools (Strumenti)

Servizi AWS

- Amazon Athena è un servizio di query interattivo che ti aiuta ad analizzare i dati direttamente in Amazon S3 utilizzando SQL standard.
- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software che aiuta a definire e fornire l'infrastruttura cloud AWS in codice.
- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account AWS e regioni AWS.
- AWS ti CloudTrail aiuta a controllare la governance, la conformità e il rischio operativo del tuo account AWS.
- AWS Glue è un servizio di estrazione, trasformazione e caricamento (ETL) completamente gestito. Ti aiuta a classificare, pulire, arricchire e spostare i dati in modo affidabile tra archivi di dati e flussi di dati. Questo modello utilizza un crawler AWS Glue e una tabella AWS Glue Data Catalog.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- Amazon Quick è un servizio di business intelligence (BI) su scala cloud che ti aiuta a visualizzare, analizzare e generare report sui dati in un'unica dashboard.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Repository di codice

Il codice per questo pattern è disponibile nel GitHub infrastructure-assessment-iac-automationrepository.

L'archivio del codice contiene i seguenti file e cartelle:

- `lib`folder — I file di costruzione in Python del CDK AWS utilizzati per creare risorse AWS
- `src/lambda_code`— Il codice Python che viene eseguito nella funzione Lambda
- `requirements.txt`— L'elenco di tutte le dipendenze Python che devono essere installate
- `cdk.json`— Il file di input per fornire i valori necessari per avviare le risorse

Best practice

Imposta il monitoraggio e gli avvisi per la funzione Lambda. Per ulteriori informazioni, consulta Monitoraggio e risoluzione dei problemi delle funzioni Lambda. Per le best practice generali relative all'utilizzo delle funzioni Lambda, consulta la documentazione AWS.

Epiche

Configurare l'ambiente

Operazione	Description	Competenze richieste
Clona il repository sul tuo computer locale.	Per clonare il repository, esegui il comando `git clone https://github.com/aws-samples/infrastructure-assessment-iac-automation.git .`	AWS DevOps, DevOps ingegnere

Configura l'ambiente virtuale Python e installa le dipendenze richieste.	Per configurare l'ambiente virtuale Python, esegui i seguenti comandi.
	`cd infrastructure-assessment-iac-automation`
	`python3 -m venv .venv`
	`source .venv/bin/activate`
		AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	Per configurare le dipendenze richieste, esegui il comando. `pip install -r requirements.txt`	
Configura l'ambiente AWS CDK e sintetizza il codice CDK AWS.	1. Per configurare l'ambiente AWS CDK nel tuo account AWS, esegui il comando `cdk bootstrap aws://ACCOUNT-NUMBER/REGION` . 2. Per convertire il codice in una configurazione CloudFormation dello stack AWS, esegui il comando `cdk synth`.	AWS DevOps, DevOps ingegnere

Configura le credenziali AWS sul tuo computer locale

Operazione	Description	Competenze richieste

Esporta le variabili per l'account e la regione in cui verrà distribuito lo stack.	Per fornire le credenziali AWS per AWS CDK utilizzando variabili di ambiente, esegui i seguenti comandi.
	`export CDK_DEFAULT_ACCOUNT=<12 Digit AWS Account Number>`
	`export CDK_DEFAULT_REGION=<region>`
		AWS DevOps, DevOps ingegnere

Configura il profilo AWS CLI.	Per configurare il profilo AWS CLI per l'account, segui le	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	istruzioni nella documenta zione AWS.	

Configura e distribuisci lo strumento di valutazione delle risorse

Operazione	Description	Competenze richieste
Distribuisci risorse nell'acco unt.	Per distribuire risorse nell'acco unt AWS utilizzando AWS CDK, procedi come segue: 1. Nella radice del repositor y clonato, nel `cdk.json` file, fornisci gli input per i seguenti parametri: • `s3_context` • `ct_context` • `kms_context` • `lambda_context` • `glue_context` • `qs_context` Questi valori definisco no le configurazioni e la nomenclatura delle risorse. I valori predefiniti sono impostati e possono essere modificati se necessario. **Note** Per evitare un errore che indica che il bucket	AWS DevOps

Operazione	Description	Competenze richieste
	S3 esiste già, assicurati di fornire nomi univoci per le `s3_contex t` sezioni `ct` and`output`. 2. Per distribuire risorse, esegui il comando. `cdk deploy` Il `cdk deploy` comando crea una CloudTrail risorsa per registrare gli eventi e salvare il file di registro nel bucket S3 di input. I file di registro del percorso verranno elaborati dalla funzione Lambda. I risultati filtrati vengono archiviat i nel bucket di output S3 e sono pronti per essere utilizzati da Amazon Athena e Amazon Quick Sight.	

Operazione	Description	Competenze richieste
Esegui il crawler AWS Glue e crea la tabella Data Catalog.	Un crawler AWS Glue viene utilizzato per mantenere dinamico lo schema dei dati. La soluzione crea e aggiorna le partizioni nella tabella del catalogo dati di AWS Glue eseguendo periodicamente il crawler come definito dallo scheduler del crawler di AWS Glue. Dopo che i dati sono disponibili nel bucket S3 di output, utilizza i seguenti passaggi per eseguire il crawler AWS Glue e creare lo schema della tabella Data Catalog per i test: 1. Accedi alla Console di gestione AWS e accedi alla console AWS Glue. 2. Nel pannello di navigazione, in Data Catalog, scegli Crawlers. 3. Seleziona il crawler. `iac-tool-qa-resource-iac-json-crawler` 4. Esegui il crawler. 5. Una volta eseguito correttamente, il crawler crea una tabella AWS Glue Data Catalog. Quick Sight utilizzerà la tabella per visualizzare i dati.	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	Note Il codice CDK AWS configura il crawler AWS Glue per l'esecuzione in un determinato momento, ma puoi anche eseguirlo su richiesta.	
Implementa il costrutto Quick Sight.	1. Per implementare il costrutto Quick Sight, decommenta il codice tra e dentro. `#QuickSight setup - start #QuickSight setup - ends` `resource_iac_tool_stack.py` 2. Dopo aver rimosso il commento, esegui il `cdk deploy` comando per creare `QuickSight DataSource` e accedere all'account Quick `QuickSight DataSet` Sight.	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
Crea la dashboard Quick Sight.	Per creare il dashboard e l'analisi Quick Sight di esempio, procedi come segue: 1. Passa alla console Quick Sight e seleziona la regione AWS in cui vengono distribuite le risorse. 2. Nel riquadro di navigazione, scegli Set di dati e verifica che un set di dati denominato sia `ct-operations-iac-ds` stato creato nel set di dati Quick Sight. Se non vedi il set di dati, ridistribuisci il costrutto Quick Sight. 3. Selezionate il **`ct-operations-iac-ds`** set di dati e scegliete USE IN ANALYSIS. 4. Seleziona il foglio predefinito. 5. Seleziona le rispettive colonne dall'elenco dei campi sul lato sinistro. 6. Dopo aver selezionato le colonne richieste, seleziona il tipo di visualizzazione appropriato per visualizzare i dati.	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	Per ulteriori informazioni, consulta Avvio di un'analisi in Amazon Quick Sight e Tipi visivi in Amazon Quick Sight.	

Pulisci tutte le risorse AWS nella soluzione

Operazione	Description	Competenze richieste
Rimuovi le risorse AWS.	1. Per rimuovere le risorse AWS distribuite dalla soluzione, esegui il comando `cdk destroy`. 2. Elimina tutti gli oggetti dai due bucket S3, quindi rimuovi i bucket. Per ulteriori informazioni, consulta Eliminazione di un bucket.	AWS DevOps, DevOps ingegnere

Configura funzionalità aggiuntive oltre all'automazione dello strumento di valutazione delle risorse AWS

Operazione	Description	Competenze richieste
Monitora e pulisci le risorse create manualmente.	(Facoltativo) Se la tua organizzazione ha requisiti di conformità per creare risorse utilizzando strumenti IaC, puoi raggiungere la conformità utilizzando l'automazione degli strumenti di valutazione delle risorse AWS per recuperare le	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	risorse assegnate manualmente. Puoi anche utilizzare lo strumento per importare le risorse in uno strumento IaC o per ricrearle. Per monitorare le risorse assegnate manualmente, esegui le seguenti attività di alto livello: 1. Implementa l'automazione degli strumenti di valutazione delle risorse AWS. 2. Imposta una funzione Lambda per interrogare quotidianamente le tabelle Athena, trovare i dati pertinenti sulle risorse assegnate manualmente ed esportarli in un file con valori separati da virgole (CSV). 3. Dopo l'esecuzione della funzione Lambda, è possibile inviare una notifica con i dati richiesti alle rispettive parti interessate. 4. Per una conservazione più lunga, il file.csv può essere archiviato nel bucket S3. 5. In base alle informazioni contenute nel file.csv, elimina le risorse create manualmente o importale in	

Operazione	Description	Competenze richieste
	una soluzione IaC esistente .	

Risoluzione dei problemi

Problema	Soluzione
AWS CDK restituisce errori.	Per assistenza con i problemi di AWS CDK, consulta Risoluzione dei problemi comuni di AWS CDK.

Risorse correlate

- Creazione di funzioni Lambda con Python
- Inizia a usare AWS CDK
- Lavorare con AWS CDK in Python
- Creazione di una traccia di log CloudTrail
- Inizia a usare Amazon Quick Sight

Informazioni aggiuntive

Account multipli

Per configurare le credenziali AWS CLI per più account, utilizza i profili AWS. Per ulteriori informazioni, consulta la sezione Configurazione di più profili in Configurazione dell'interfaccia a riga di comando di AWS.

Comandi AWS CDK

Quando lavori con AWS CDK, tieni presente i seguenti comandi utili:

- Elenca tutti gli stack presenti nell'app

```
cdk ls
```

- Emette il modello AWS sintetizzato CloudFormation

```
cdk synth
```

- Distribuisce lo stack nell'account e nella regione AWS predefiniti

```
cdk deploy
```

- Confronta lo stack distribuito con lo stato attuale

```
cdk diff
```

- Apre la documentazione di AWS CDK

```
cdk docs
```

Installa automaticamente i sistemi SAP utilizzando strumenti open source

Guilherme Sesterheim, Amazon Web Services

Riepilogo

Questo modello mostra come automatizzare l'installazione dei sistemi SAP utilizzando strumenti open source per creare le seguenti risorse:

- Un database SAP S/4HANA 1909
- Un'istanza SAP ABAP Central Services (ASCS)
- Un'istanza SAP Primary Application Server (PAS)

HashiCorp Terraform crea l'infrastruttura del sistema SAP e Ansible configura il sistema operativo (OS) e installa le applicazioni SAP. Jenkins esegue l'installazione.

Questa configurazione trasforma l'installazione dei sistemi SAP in un processo ripetibile, che può contribuire ad aumentare l'efficienza e la qualità dell'implementazione.

> **Note**
>
> Il codice di esempio fornito in questo modello funziona sia per i sistemi ad alta disponibilità (HA) che per i sistemi non HA.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un bucket Amazon Simple Storage Service (Amazon S3) che contiene tutti i tuoi file multimediali SAP
- Un principal AWS Identity and Access Management (IAM) con una chiave di accesso e una chiave segreta e che dispone delle seguenti autorizzazioni:
  - Autorizzazioni di sola lettura: Amazon Route 53, AWS Key Management Service (AWS KMS)
  - Autorizzazioni di lettura e scrittura: Amazon S3, Amazon Elastic Compute Cloud (Amazon), EC2 Amazon Elastic File System (Amazon EFS), IAM, Amazon, Amazon CloudWatch DynamoDB

- Una zona ospitata privata sulla Route 53
- Un abbonamento a Red Hat Enterprise Linux per SAP con HA e Update Services 8.2 Amazon Machine Image (AMI) in Amazon Marketplace
- Una chiave AWS KMS gestita dal cliente
- Una coppia di key pair Secure Shell (SSH)
- Un gruppo EC2 di sicurezza Amazon che consente la connessione SSH sulla porta 22 dal nome host su cui si installa Jenkins (il nome host è molto probabilmente localhost)
- Vagrant viene installato e configurato HashiCorp
- VirtualBoxdi Oracle installato e configurato
- Familiarità con Git, Terraform, Ansible e Jenkins

Limitazioni

- Solo SAP S/4HANA 1909 è stato completamente testato per questo scenario specifico. Il codice Ansible di esempio in questo modello richiede una modifica se si utilizza un'altra versione di SAP HANA.
- La procedura di esempio riportata in questo modello funziona per i sistemi operativi Mac OS e Linux. Alcuni comandi possono essere eseguiti solo su terminali basati su UNIX. Tuttavia, è possibile ottenere un risultato simile utilizzando comandi diversi e un sistema operativo Windows.

Versioni del prodotto

- SAP S/4HANA 1909
- Red Hat Enterprise Linux (RHEL) 8.2 o versioni successive

Architecture

Il diagramma seguente mostra un esempio di flusso di lavoro che utilizza strumenti open source per automatizzare l'installazione dei sistemi SAP in un account AWS:

Il diagramma mostra il flusso di lavoro seguente:

1. Jenkins orchestra l'esecuzione dell'installazione del sistema SAP eseguendo il codice Terraform e Ansible.

2. Il codice Terraform crea l'infrastruttura del sistema SAP.
3. Il codice Ansible configura il sistema operativo e installa le applicazioni SAP.
4. Un database SAP S/4HANA 1909, un'istanza ASCS e un'istanza PAS che includono tutti i prerequisiti definiti vengono installati su un'istanza Amazon. EC2

> **Note**
>
> La configurazione di esempio in questo modello crea automaticamente un bucket Amazon S3 nel tuo account AWS per archiviare il file di stato Terraform.

Stack tecnologico

- Terraform
- Ansible
- Jenkins
- Un database SAP S/4HANA 1909
- Un'istanza SAP ASCS
- Un'istanza SAP PAS
- Amazon EC2

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e scalarli rapidamente verso l'alto o verso il basso.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito. Questa rete virtuale è simile a una rete tradizionale che gestiresti nel tuo data center, con i vantaggi dell'utilizzo dell'infrastruttura scalabile di AWS.

Altri strumenti

- HashiCorp Terraform è un'applicazione di interfaccia a riga di comando che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud.
- Ansible è uno strumento open source di configurazione come codice (CaC) che aiuta ad automatizzare applicazioni, configurazioni e infrastrutture IT.
- Jenkins è un server di automazione open source che consente agli sviluppatori di creare, testare e distribuire il proprio software.

Codice

Il codice per questo pattern è disponibile nel repository -jenkins-ansible. GitHub aws-install-sap-with

Epiche

Configura i prerequisiti

Operazione	Description	Competenze richieste
Aggiungi i tuoi file multimediali SAP a un bucket Amazon S3.	Crea un bucket Amazon S3 che contenga tutti i tuoi file multimediali SAP. **Important** Assicurati di seguire la gerarchia delle cartelle di AWS Launch Wizard per S/4HANA nella documentazione di Launch Wizard.	Amministratore del cloud
Installa VirtualBox.	Installazione e configurazione VirtualBoxtramite Oracle.	DevOps ingegnere
Installa Vagrant.	Installa e configura Vagrant tramite. HashiCorp	DevOps ingegnere

Operazione	Description	Competenze richieste
Configura il tuo account AWS.	1. Verifica di disporre di un preside IAM con una chiave di accesso e una chiave segreta e che disponga delle seguenti autorizza zioni: • Autorizzazioni di sola lettura: Amazon Route 53, AWS Key Management Service (AWS KMS) • Autorizzazioni di lettura e scrittura: Amazon S3, Amazon Elastic Compute Cloud (Amazon), EC2 Amazon Elastic File System (Amazon EFS), IAM, Amazon, Amazon CloudWatch DynamoDB 2. Salva la chiave di accesso e la chiave segreta del principale IAM per consultar la in un secondo momento. 3. Crea una zona ospitata privata Route 53, se non ne hai già una. Salva il nome della zona (ad esempio, sapteam.net) per riferimen to successivo. 4. Abbonati all'AMI Red Hat Enterprise Linux per SAP con HA e Update Services 8.2 in Amazon Marketplace.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	Salva l'ID AMI (ad esempio, ami-0000000) per riferimento successivo. 5. Crea una chiave AWS KMS gestita dal cliente. Salva l'Amazon Resource Name (ARN) della chiave KMS per riferimento successivo. **Note** Di seguito è riportato un esempio di ARN della chiave gestita dai clienti AWS KMS: arn:aws:kms:us-east- 1:123412341234:key/uuid 6. Crea una coppia di chiavi SSH. Salva il nome della coppia di chiavi e il file.pem per riferimento successivo. 7. Crea un gruppo EC2 di sicurezza Amazon che consenta la connessione SSH sulla porta 22 dal nome host in cui installi Jenkins. Salva l'ID del gruppo di sicurezza per riferimento successivo.	

Operazione	Description	Competenze richieste
	Note Il nome host è molto probabilmente localhost.	

Crea ed esegui la tua installazione SAP

Operazione	Description	Competenze richieste
Clona il repository di codice da. GitHub	Clona il repository aws-installl-sap-with-jenkins-ansible su. GitHub	DevOps ingegnere
Avvia il servizio Jenkins.	Apri il terminale Linux. Quindi, vai alla cartella locale che contiene la cartella del repository del codice clonato ed esegui il seguente comando: `sudo vagrant up` **Note** L'avvio di Jenkins richiede circa 20 minuti. In caso di esito positivo, il comando restituisce un messaggio che	DevOps ingegnere

Operazione	Description	Competenze richieste
	indica che il servizio è attivo e funzionante.	
Apri Jenkins in un browser web e accedi.	1. In un browser Web, inserisci http://localhost:5555. Jenkins si apre. 2. Accedi a Jenkins utilizzando admin per il nome utente e my_secret_pass_from_vault per la password.	DevOps ingegnere

Operazione	Description	Competenze richieste
Configura i parametri di installazione del sistema SAP.	1. In Jenkins, scegli Gestisci Jenkins. Quindi, scegli Gestisci credenziali. Viene visualizzato un elenco di variabili di credenziali che puoi configurare. 2. Configura tutte le seguenti variabili di credenziali: • Per AWS_ACCOUNT_CREDENTIALS, inserisci l'ID della chiave di accesso e l'ID della chiave di accesso segreta del tuo principale IAM. • Per AMI_ID, inserite l'ID AMI di Red Hat Enterprise Linux for SAP with HA and Update Services 8.2 AMI. • Per KMS_KEY_ARN, inserisci l'ARN della chiave gestita dal cliente AWS KMS. • Per SSH_KEYPAIR_NAME, inserisci il nome della tua coppia di chiavi SSH, senza inserire il tipo di file.pem. • Per SSH_KEYPAIR_FILE, inserisci il nome completo del file.pem della tua coppia di chiavi (ad esempio, mykeypair.pem). Assicurati di caricare anche il file.pem	Amministratore di sistema AWS, DevOps ingegnere

Operazione	Description	Competenze richieste
	delle coppie di chiavi su Jenkins. • Per S3_ROOT_F OLDER_INSTALL_FILES, inserisci il nome del bucket Amazon S3 e della cartella, se applicabile, (ad esempio, s3:///S4H1909) che contiene i tuoi file multimediali SAP. my-media-bucket • Per PRIVATE_D NS_ZONE_NAME, inserisci il nome della tua zona ospitata privata sulla Route 53 (ad esempio, myprivate companyurl.net). • Per VPC_ID, inserisci l'ID VPC (ad esempio, vpc-12345) dell'Amazon VPC in cui stai creando le risorse SAP. • Per SUBNET_IDS, inserisci due sottoreti pubbliche IDs se lavori in un ambiente di test (per le future funzional ità HA). Se lavori in un ambiente di produzione, è consigliabile utilizzare due sottoreti private con un host bastion. • Per SECURITY_ GROUP_ID, inserisci l'ID del gruppo di EC2 sicurezza Amazon che consente la	

Operazione	Description	Competenze richieste
	connessione SSH sulla porta 22 dal nome host su cui hai installato Jenkins. **Note** Puoi configurare gli altri parametri non richiesti secondo necessità, in base al tuo caso d'uso. Ad esempio, puoi modificare l'ID di sistema SAP (SID) delle istanze, la password, i nomi e i tag predefiniti per il tuo sistema SAP. Tutte le variabili obbligatorie hanno (Obbligatorio) all'inizio dei loro nomi.	

Operazione	Description	Competenze richieste
Esegui l'installazione del tuo sistema SAP.	1. In Jenkins, scegli Jenkins Home. Quindi, scegli SAP HANA+ASCS+PAS 3 Instances. 2. Scegli Spin up e installa. Quindi, scegli Main. 3. Scegli Costruisci ora. Per informazioni sulle fasi della pipeline, consulta la sezione Comprendere le fasi della pipeline di Automatizzare l'installazione di SAP con strumenti open source sul blog di AWS. **Note** Se si verifica un errore, sposta il cursore sulla casella di errore rossa che appare e scegli Logs. Vengono visualizzati i log relativi alla fase della pipeline che ha generato un errore. La maggior parte degli errori si verifica a causa di impostazioni errate dei parametri.	DevOps ingegnere, amministratore di sistema AWS

Risorse correlate

- DevOps per SAP — Installazione SAP: da 2 mesi a 2 ore (DevOps Enterprise Summit Video Library)

Automatizza il portafoglio e la distribuzione dei prodotti di AWS Service Catalog utilizzando AWS CDK

Sandeep Gawande, Viyoma Sachdeva e RAJNEESH TYAGI, Amazon Web Services

Riepilogo

AWS Service Catalog ti aiuta a gestire centralmente i cataloghi di servizi o prodotti IT approvati per l'uso nell'ambiente AWS della tua organizzazione. Una raccolta di prodotti è denominata portfolio e un portfolio contiene anche informazioni di configurazione. Con AWS Service Catalog, puoi creare un portafoglio personalizzato per ogni tipo di utente della tua organizzazione e quindi concedere l'accesso al portafoglio appropriato. Questi utenti possono quindi distribuire rapidamente qualsiasi prodotto di cui hanno bisogno all'interno del portafoglio.

Se si dispone di un'infrastruttura di rete complessa, ad esempio architetture multiregionali e multi-account, si consiglia di creare e gestire i portafogli Service Catalog in un unico account centrale. Questo modello descrive come utilizzare AWS Cloud Development Kit (AWS CDK) per automatizzare la creazione di portafogli Service Catalog in un account centrale, concedere agli utenti finali l'accesso ad essi e quindi, facoltativamente, fornire prodotti in uno o più account AWS target. Questa ready-to-use soluzione crea i portafogli Service Catalog nell'account di origine. Inoltre, facoltativamente, effettua il provisioning dei prodotti negli account di destinazione utilizzando gli CloudFormation stack AWS e aiuta a configurare TagOptions i prodotti:

- AWS CloudFormation StackSets: puoi StackSets utilizzarlo per lanciare prodotti Service Catalog su più regioni e account AWS. In questa soluzione, hai la possibilità di effettuare il provisioning automatico dei prodotti quando la distribuisci. Per ulteriori informazioni, consulta Using AWS CloudFormation StackSets (Service Catalog documentation) e StackSets concepts (CloudFormation documentation).
- TagOption libreria: puoi gestire i tag sui prodotti forniti utilizzando la TagOption libreria. A TagOptionè una coppia chiave-valore gestita in AWS Service Catalog. Non è un tag AWS, ma funge da modello per creare un tag AWS basato su TagOption. Per ulteriori informazioni, vedere TagOption libreria (documentazione del Service Catalog).

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo che desideri utilizzare come account di origine per l'amministrazione dei portafogli Service Catalog.
- Se utilizzi questa soluzione per fornire prodotti in uno o più account di destinazione, l'account di destinazione deve già esistere ed essere attivo.
- Autorizzazioni AWS Identity and Access Management (IAM) per accedere ad AWS Service Catalog CloudFormation, AWS e AWS IAM.

Versioni del prodotto

- CDK AWS versione 2.27.0

Architecture

Stack tecnologico Target

- Portafogli Service Catalog in un account AWS centralizzato
- Prodotti Service Catalog distribuiti nell'account di destinazione

Architettura Target

1. Nell'account portfolio (o source), aggiorni il file config.json con l'account AWS, la regione AWS, il ruolo IAM, il portafoglio e le informazioni sul prodotto per il tuo caso d'uso.
2. Distribuisci l'applicazione AWS CDK.
3. L'applicazione AWS CDK assume il ruolo IAM di implementazione e crea i portafogli e i prodotti Service Catalog definiti nel file config.json.

   Se hai configurato StackSets per distribuire prodotti in un account di destinazione, il processo continua. Se non hai configurato alcun prodotto StackSets per fornire alcun prodotto, il processo è completo.
4. L'applicazione AWS CDK assume il ruolo di StackSet amministratore e distribuisce lo CloudFormation stack set AWS definito nel file config.json.
5. Nell'account di destinazione, StackSets assume il ruolo di esecuzione ed effettua il provisioning dei prodotti. StackSet

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software che aiuta a definire e fornire l'infrastruttura cloud AWS in codice.
- AWS CDK Toolkit è un kit di sviluppo cloud a riga di comando che ti aiuta a interagire con la tua app AWS CDK.
- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account e regioni AWS.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Service Catalog ti aiuta a gestire centralmente i cataloghi di servizi IT approvati per AWS. Gli utenti finali possono distribuire rapidamente soltanto i servizi IT approvati di cui hanno bisogno, in accordo con i vincoli stabiliti dall'organizzazione.

Repository di codice

Il codice per questo pattern è disponibile su GitHub, nel aws-cdk-servicecatalog-automationrepository. L'archivio del codice contiene i seguenti file e cartelle:

- cdk-sevicecatalog-app— Questa cartella contiene l'applicazione AWS CDK per questa soluzione.
- config: questa cartella contiene il file config.json e il CloudFormation modello per la distribuzione dei prodotti nel portafoglio Service Catalog.
- config/config.json: questo file contiene tutte le informazioni di configurazione. Aggiorna questo file per personalizzare questa soluzione in base al tuo caso d'uso.
- config/templates: questa cartella contiene i CloudFormation modelli per i prodotti Service Center.
- setup.sh: questo script distribuisce la soluzione.
- uninstall.sh: questo script elimina lo stack e tutte le risorse AWS create durante la distribuzione di questa soluzione.

Per utilizzare il codice di esempio, segui le istruzioni nella sezione Epics.

Best practice

- I ruoli IAM utilizzati per implementare questa soluzione devono rispettare il principio del privilegio minimo (documentazione IAM).
- Aderisci alle migliori pratiche per lo sviluppo di applicazioni cloud con AWS CDK (post sul blog AWS).
- Rispetta le CloudFormation best practice di AWS (CloudFormation documentazione).

Epiche

Configurare l'ambiente

Operazione	Description	Competenze richieste

Installa AWS CDK Toolkit.	Assicurati di avere installato AWS CDK Toolkit. Inserisci il seguente comando per confermare se è installato e verificare la versione.
	`cdk --version`
	Se AWS CDK Toolkit non è installato, inserisci il seguente comando per installarlo.
	`npm install -g aws-cdk@2.27.0`
	Se la versione di AWS CDK Toolkit è precedente alla 2.27.0, immetti il seguente comando per aggiornarla alla versione 2.27.0.
		AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	`npm install -g aws-cdk@2.27.0 --force`	

Clonare il repository.	Inserire il seguente comando. In Clona il repository nella sezione Informazioni aggiuntive, puoi copiare il comando completo contenente l'URL del repository. Questo clona il repository da. aws-cdk-servicecatalog-automation GitHub
	`git clone <repository-URL>.git`
	Questo crea una `cd aws-cdk-servicecatalog-automation` cartella nella directory di destinazione. Immettete il seguente comando per navigare in questa cartella.
	`cd aws-cdk-servicecatalog-automation`
		AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste

Configura le credenziali AWS.	Esegui i comandi seguenti: Questi esportano le seguenti variabili, che definiscono l'account AWS e la regione in cui stai distribuendo lo stack.
	`export CDK_DEFAU LT_ACCOUNT=<12-digit AWS account number>`
	`export CDK_DEFAU LT_REGION=<AWS Region>`
	Le credenziali AWS per AWS CDK vengono fornite tramite variabili di ambiente.
		AWS DevOps, DevOps ingegnere

Configura le autorizzazioni per i ruoli IAM degli utenti finali.	Se intendi utilizzare i ruoli IAM per concedere l'accesso al portafoglio e ai prodotti in esso contenuti, i ruoli devono disporre delle autorizzazioni che devono essere assunte dal responsabile del servizio servicecatalog.amazonaws.com. Per istruzioni su come concedere queste autorizzazioni, consulta Enabling trusted access with Service Catalog (documentazione di AWS Organizations).	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
Configura i ruoli IAM richiesti da StackSets.	Se utilizzi StackSets il provisioning automatico dei prodotti negli account di destinazione, devi configurare i ruoli IAM che amministrano ed eseguono il set di stack. 1. Nell'account di origine, conferma se esiste `AWSCloudFormationStackSetAdministrationRole` già. Negli account di destinazione, conferma se esiste `AWSCloudFormationStackSetExecutionRole` già. Se questi ruoli esistono già, puoi passare all'epopea successiva. 2. Segui le istruzioni in Grant self-managed permissions (documentazione IAM) per creare il ruolo di amministrazione dello stack set nell'account di portafoglio e creare il ruolo di esecuzione in ogni account di destinazione.	AWS DevOps, DevOps ingegnere

Personalizza e distribuisci la soluzione

Operazione	Description	Competenze richieste
Crea i CloudFormation modelli.	Nella `config/templates` cartella, crea CloudForm ation modelli per tutti i prodotti che desideri includere nei tuoi portafogli. Per ulteriori informazioni, consulta Working with AWS CloudFormation templates (CloudFormation documentazione).	Sviluppatore di app, AWS DevOps, DevOps ingegnere
Personalizza il file di configura zione.	Nella `config` cartella, apri il file config.json e definisci i parametri appropriati per il tuo caso d'uso. Salvo diversa indicazione, sono obbligatori i seguenti parametri: Nella `portfolios` sezione, definite i seguenti parametri per creare uno o più portafogli Service Catalog: • `portfolioName` — Il nome del portfolio. • `providerName` — Il nome della persona, del team o dell'organizzazione che gestisce il portfolio. • `description` — Una breve descrizione del portafoglio. • `roles`— (Facoltativo) Nomi di tutti i ruoli IAM che	Sviluppatore di app, DevOps ingegnere, AWS DevOps

Operazione	Description	Competenze richieste
	dovrebbero avere accesso a questo portafoglio. Gli utenti che hanno questo ruolo possono accedere ai prodotti di questo portafogl io. • `users`— (Facoltativo) Nomi di tutti gli utenti IAM che dovrebbero avere accesso a questo portafoglio e ai suoi prodotti. • `groups`— (Facoltativo) Nomi di tutti i gruppi di utenti IAM che dovrebbero avere accesso a questo portafogl io e ai suoi prodotti. **Warning** Gli utenti IAM dispongono di credenziali a lungo termine, il che rappresenta un rischio per la sicurezza. Per ridurre questo rischio, si consiglia di fornire a questi utenti solo le autorizza zioni necessarie per eseguire l'attività e di rimuoverli quando non sono più necessari.	

Operazione	Description	Competenze richieste

	Important
	`rolesusers`, e `groups` sono tutti parametri opzionali , ma se non si definisce uno di questi parametri, nessuno potrà visualizzare i prodotti del portafogl io nella console Service Catalog. Definite almeno uno di questi parametri. Per ulteriori informazi oni, vedere Concedere le autorizzazioni agli utenti finali di Service Catalog (document azione di Service Catalog).
	
	• (Facoltativo) Nella `tagOption` sezione, definisci TagOptions per i prodotti:
	  • `key`— Nome della TagOption chiave
	  • `value`— Valori di stringa consentiti per TagOption
	
	Per ulteriori informazioni, vedere TagOption libreria	

Operazione	Description	Competenze richieste
	(documentazione del Service Catalog). • Nella **`products`**sezione, definisci i seguenti parametri per i prodotti: • `portfolioName` — Il nome del portafoglio in cui desideri aggiunger e il prodotto. È possibile specificare un solo portafoglio. • `productName` — Il nome del prodotto. • `owner`— Il proprietario del prodotto. • `productVersionName` — Il nome della versione del prodotto in formato stringa, ad esempio`v1`. • `templatePath` — Il percorso del file per il CloudFormation modello per il prodotto. • `deployWithStackSet s` — (Facoltativo) Specificate uno o più account e regioni da utilizzare StackSets per il provisioning automatico dei prodotti nei portafogl i. Se si utilizza questa opzione di distribuzione, sono necessari tutti i	

Operazione	Description	Competenze richieste
	seguenti parametri in questa sezione: • `accounts`— Gli account di destinazione. • `regions`— Le regioni bersaglio. • `stackSetAdministrationRoleName` — Il nome del ruolo IAM utilizzato per amministrare la StackSets configurazione. Non modificare questo valore. Questo ruolo deve avere questo nome esatto. • `stackSetExecutionRoleName` — Il nome del ruolo IAM nell'account di destinazione che distribuisce le istanze dello stack. Non modificare questo valore. Questo ruolo deve avere questo nome esatto. Per un esempio di file di configurazione completato, consulta File di configura	

Operazione	Description	Competenze richieste
	zione di esempio nella sezione Informazioni aggiuntive.	
Distribuire la soluzione.	Inserire il seguente comando. Questo distribuisce l'app AWS CDK ed effettua il provisioning dei portafogli e dei prodotti Service Catalog come specificato nel file config.json. `sh +x setup.sh`	Sviluppatore di app, DevOps ingegnere, AWS DevOps

Operazione	Description	Competenze richieste
Verifica la distribuzione.	Verifica la corretta implement azione effettuando le seguenti operazioni: 1. Accedi alla Console di gestione AWS con credenziali che possono accedere a uno o più portafogli definiti nel file di configurazione. 2. Aprire la console Service Catalog all'indirizzo https://console.aws.amazon.com/servicecatalog/. 3. Nel riquadro di navigazione, in Provisioning, scegli Prodotti. Verifica di visualizzare un elenco di prodotti che hai specificato per il portafoglio. 4. Segui le istruzioni riportate in Launching a product (documentazione Service Catalog) per lanciare uno dei prodotti disponibili. Verifica che le versioni e i tag del prodotto disponibili corrispondano ai valori forniti nel file di configurazione. 5. Se hai scelto di fornire automaticamente i prodotti in uno o più account di destinazione utilizzando	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	StackSets, procedi come segue: a. Accedi con le credenzia li che ti consentono di visualizzare i prodotti forniti in uno degli account di destinazione. b. Nella console Service Catalog, nel pannello di navigazione, in Provision ing, scegli Provisioned products. c. Verifica che i prodotti previsti compaiano nell'elenco.	

Operazione	Description	Competenze richieste
(Facoltativo) Aggiorna i portafogli e i prodotti.	Se desideri utilizzare questa soluzione per aggiornare i portafogli o i prodotti o per fornire nuovi prodotti: 1. Apporta le modifiche richieste nel file config.json. 2. Aggiungete o modificate CloudFormation i modelli necessari nella cartella. `config/template` 3. Ridistribuisci la soluzione. Ad esempio, puoi aggiungere portafogli aggiuntivi o fornire più risorse. L'app AWS CDK implementa solo le modifiche . Se non ci sono modifiche ai portafogli o ai prodotti precedentemente distribuiti, la ridistribuzione non li influenza.	Sviluppatore di app, DevOps ingegnere, General AWS

Pulisci la soluzione

Operazione	Description	Competenze richieste
(Facoltativo) Rimuovi le risorse AWS distribuite da questa soluzione.	Se desideri eliminare un prodotto fornito, segui le istruzioni in Eliminazione dei prodotti forniti (document azione del Service Catalog). Se desideri eliminare tutte le risorse create da questa	AWS DevOps, DevOps ingegnere, sviluppatore di app

Operazione	Description	Competenze richieste
	soluzione, inserisci il seguente comando. `sh uninstall.sh`	

Risorse correlate

- Libreria AWS Service Catalog Construct (riferimento alle API AWS)
- StackSets concetti (CloudFormation documentazione)
- AWS Service Catalog (marketing AWS)
- Utilizzo di Service Catalog con AWS CDK (AWS workshop)

Informazioni aggiuntive

Clonare il repository

Immettere il seguente comando da cui clonare il repository. GitHub

```
git clone https://github.com/aws-samples/aws-cdk-servicecatalog-automation.git
```

File di configurazione di esempio

Di seguito è riportato un file config.json di esempio con valori di esempio.

```
{
    "portfolios": [
        {
            "displayName": "EC2 Product Portfolio",
            "providerName": "User1",
            "description": "Test1",
            "roles": [
                "<Names of IAM roles that can access the products>"
            ],
            "users": [
                "<Names of IAM users who can access the products>"
            ],
            "groups": [
```

```
                "<Names of IAM user groups that can access the products>"
            ]
        },
        {
            "displayName": "Autoscaling Product Portfolio",
            "providerName": "User2",
            "description": "Test2",
            "roles": [
                "<Name of IAM role>"
            ]
        }
    ],
    "tagOption": [
        {
            "key": "Group",
            "value": [
                "finance",
                "engineering",
                "marketing",
                "research"
            ]
        },
        {
            "key": "CostCenter",
            "value": [
                "01",
                "02",
                "03",
                "04"
            ]
        },
        {
            "key": "Environment",
            "value": [
                "dev",
                "prod",
                "stage"
            ]
        }
    ],
    "products": [
        {
            "portfolioName": "EC2 Product Profile",
            "productName": "Ec2",
```

```
            "owner": "owner1",
            "productVersionName": "v1",
            "templatePath": "../../config/templates/template1.json"
        },
        {
            "portfolioName": "Autoscaling Product Profile",
            "productName": "autoscaling",
            "owner": "owner1",
            "productVersionName": "v1",
            "templatePath": "../../config/templates/template2.json",
            "deployWithStackSets": {
                "accounts": [
                    "012345678901",
                ],
                "regions": [
                    "us-west-2"
                ],
                "stackSetAdministrationRoleName":
 "AWSCloudFormationStackSetAdministrationRole",
                "stackSetExecutionRoleName": "AWSCloudFormationStackSetExecutionRole"
            }
        }
    ]
}
```

Automatizza la gestione dinamica delle pipeline per l'implementazione di soluzioni hotfix in ambienti Gitflow utilizzando e AWS Service Catalog AWS CodePipeline

Balaji Vedagiri, Faisal Shahdad, Shanmugam Shanker e Vivek Thangamuthu, Amazon Web Services

Riepilogo

> **Note**
>
> AWS CodeCommit non è più disponibile per i nuovi clienti. I clienti esistenti di AWS CodeCommit possono continuare a utilizzare il servizio normalmente. Ulteriori informazioni

Questo modello affronta uno scenario di gestione di una pipeline dinamica di hotfix dedicata esclusivamente alla distribuzione sicura di soluzioni hotfix in un ambiente di produzione. La soluzione viene implementata e gestita utilizzando un portafoglio e un prodotto. AWS Service Catalog Per l'automazione degli eventi viene utilizzata una EventBridge regola Amazon. Le restrizioni vengono applicate utilizzando i vincoli del portafoglio di Service Catalog e i ruoli AWS Identity and Access Management (IAM) per gli sviluppatori. È consentita solo una AWS Lambda funzione per avviare il prodotto Service Catalog, attivata dalla EventBridge regola. Questo modello è progettato per ambienti con una configurazione Gitflow specifica, descritta in Informazioni aggiuntive.

In genere, viene implementato un hotfix per risolvere problemi critici o di sicurezza segnalati in un ambiente live, come Production. Gli hotfix devono essere implementati direttamente solo negli ambienti di staging e produzione. Le pipeline di staging e production sono ampiamente utilizzate per le richieste di sviluppo regolari. Queste pipeline non possono essere utilizzate per implementare hotfix perché esistono funzionalità continue di garanzia della qualità che non possono essere promosse alla produzione. Per rilasciare gli hotfix, questo schema descrive una pipeline dinamica e di breve durata con le seguenti funzionalità di sicurezza:

- Creazione automatica: una pipeline di hotfix viene creata automaticamente ogni volta che viene creato un ramo di hotfix in un repository. AWS CodeCommit
- Restrizioni di accesso: gli sviluppatori non hanno accesso alla creazione di questa pipeline al di fuori del processo di hotfix.
- Fase controllata: la pipeline dispone di una fase controllata con uno speciale token di accesso, che garantisce che una pull request (PR) possa essere creata una sola volta.

- Fasi di approvazione: le fasi di approvazione sono incluse nella pipeline per ottenere le approvazioni necessarie dalle parti interessate.
- Eliminazione automatica: la pipeline degli hotfix viene eliminata automaticamente ogni volta che un `hotfix` ramo viene eliminato dal CodeCommit repository dopo l'unione con un PR.

Prerequisiti e limitazioni

Prerequisiti

- Ne sono necessari tre attivi come Account AWS segue:
  - Account Tools: per l'integrazione continua e la configurazione della distribuzione continua (CI/CD).
  - Account Stage: per i test di accettazione da parte degli utenti.
  - Account di produzione: per un utente finale aziendale.
  - (Facoltativo) Aggiungi un account Account AWS che funga da account di controllo qualità. Questo account è necessario se si desidera sia una configurazione della pipeline principale, incluso il controllo qualità, sia una soluzione di pipeline hotfix per i test.
- Uno AWS CloudFormation stack con una condizione opzionale da distribuire nell'account QA utilizzando la pipeline principale, se necessario. Il pattern può ancora essere testato senza la configurazione della pipeline principale creando ed eliminando un ramo. `hotfix`
- Un bucket Amazon Simple Storage Service (Amazon S3) per archiviare CloudFormation i modelli utilizzati per creare prodotti Service Catalog.
- Crea regole di approvazione PR per il CodeCommit repository in conformità ai requisiti di conformità (dopo aver creato il repository).
- Limita le autorizzazioni IAM degli sviluppatori e dei responsabili del team per negare l'esecuzione della funzione prcreation-lambda Lambda perché dovrebbe essere richiamata solo dalla pipeline.

Limitazioni

- Il CloudFormation provider viene utilizzato nella fase di implementazione e l'applicazione viene distribuita utilizzando un set di modifiche. CloudFormation Se desideri utilizzare un'opzione di distribuzione diversa, modifica lo CodePipeline stack come richiesto.
- Questo modello utilizza AWS CodeBuild e altri file di configurazione per distribuire un microservizio di esempio. Se hai un tipo di carico di lavoro diverso (ad esempio, carichi di lavoro serverless), devi aggiornare tutte le configurazioni pertinenti.

- Questo modello distribuisce l'applicazione in un'unica soluzione Regione AWS (ad esempio, US East (Virginia settentrionale) us-east-1). Account AWS Per la distribuzione in più regioni, modifica il riferimento alla regione nei comandi e negli stack.
- Alcune Servizi AWS non sono disponibili in tutte. Regioni AWS Per la disponibilità regionale, consulta i servizi AWS per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Architecture

I diagrammi di questa sezione forniscono i flussi di lavoro per un evento del ciclo di vita di creazione e per un evento del ciclo di vita di eliminazione.

Il diagramma precedente per la creazione di un evento del ciclo di vita mostra quanto segue:

1. Lo sviluppatore crea una `hotfix-*` filiale nel CodeCommit repository per sviluppare una soluzione relativa all'hotfix.
2. L'evento di creazione del `hotfix-*` ramo viene acquisito tramite la regola. EventBridge I dettagli dell'evento includono il nome del repository e il nome del ramo.
3. La EventBridge regola richiama la funzione. AWS Lambda `hotfix-lambda-function` La EventBridge regola passa le informazioni sull'evento alla funzione Lambda come input.
4. La funzione Lambda elabora l'input per recuperare il nome del repository e il nome del ramo. Avvia il prodotto Service Catalog con i valori recuperati dall'input elaborato.
5. Il prodotto Service Catalog include una configurazione della pipeline che implementerà la soluzione negli ambienti Stage e Production. Il blocco pipeline include le fasi di origine, compilazione e distribuzione. Inoltre, è prevista una fase di approvazione manuale per promuovere l'implementazione per l'ambiente di produzione.
6. La fase di origine recupera il codice dal repository e dal `hotfix-*` ramo creati nella prima fase. Il codice viene passato alla fase di compilazione tramite un bucket Amazon S3 per gli artefatti. Nella fase di creazione, viene creata un'immagine del contenitore che include l'hotfix sviluppato nella `hotfix-*` filiale e inserito in Amazon Elastic Container Registry (Amazon ECR) Elastic Container Registry (Amazon ECR).
7. La fase di distribuzione sull'ambiente di fase aggiorna Amazon Elastic Container Service (Amazon ECS) con l'immagine del contenitore più recente che include l'hotfix. L'hotfix viene distribuito creando ed eseguendo un set di modifiche. CloudFormation

8. La funzione `prcreation-lambda` Lambda viene richiamata dopo una corretta implementazione nell'ambiente Stage. Questa funzione Lambda crea un PR dal `hotfix-*` ramo ai `main` rami `develop` e del repository. La funzione Lambda assicura che la correzione sviluppata nel `hotfix-*` ramo venga ricongiunta e inclusa nelle distribuzioni successive.
9. Una fase di approvazione manuale aiuta a garantire che le parti interessate necessarie esaminino la correzione e approvino l'implementazione in produzione.
10. La fase di distribuzione nell'ambiente di produzione aggiorna Amazon ECS con l'immagine del contenitore più recente che include l'hotfix. L'hotfix viene distribuito creando ed eseguendo un set di modifiche. CloudFormation

Il diagramma precedente per l'eliminazione di un evento del ciclo di vita mostra quanto segue:

1. Lo sviluppatore elimina il `hotfix-*` ramo dopo la corretta implementazione dell'hotfix nell'ambiente di produzione.
2. L'evento di eliminazione del `hotfix-*` ramo viene acquisito tramite una EventBridge regola. I dettagli dell'evento includono il nome del repository e il nome del ramo.
3. La EventBridge regola richiama la funzione Lambda. La EventBridge regola passa le informazioni sull'evento alla funzione Lambda come input.
4. La funzione Lambda elabora l'input per recuperare il nome del repository e il nome del ramo. La funzione Lambda determina il rispettivo prodotto Service Catalog dall'input passato e quindi termina il prodotto.
5. La cessazione del prodotto fornito da Service Catalog elimina la pipeline e le risorse pertinenti create in precedenza in quel prodotto.

Automazione e scalabilità

- Il modello include una EventBridge regola e una funzione Lambda, in grado di gestire più richieste di creazione di branch hotfix in parallelo. La funzione Lambda fornisce al prodotto Service Catalog la regola dell'evento corrispondente.
- La configurazione della pipeline viene gestita utilizzando il prodotto Service Catalog, che fornisce funzionalità di controllo della versione. La soluzione è inoltre scalabile automaticamente per gestire più sviluppi di hotfix per la stessa applicazione in parallelo.

- La funzione prcreation-lambda assicura che queste modifiche all'hotfix vengano reintegrate anche nei branch `main` e nei branch tramite la creazione automatica di una richiesta pull. `develop` Questo approccio è essenziale per mantenere aggiornate le `main` e le `develop` branch con tutte le correzioni ed evitare potenziali regressioni del codice. Questo processo aiuta a mantenere la coerenza tra i rami e a prevenire le regressioni del codice assicurando che tutti i rami di lunga durata dispongano delle correzioni più recenti.

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita tra e. Account AWS Regioni AWS
- AWS CodeBuildè un servizio di compilazione completamente gestito che consente di compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per l'implementazione.
- AWS CodeCommitè un servizio di controllo delle versioni che consente di archiviare e gestire in modo privato gli archivi Git, senza dover gestire il proprio sistema di controllo del codice sorgente. AWS CodeCommit non è più disponibile per i nuovi clienti. I clienti esistenti di AWS CodeCommit possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, vedi Come migrare il tuo AWS CodeCommit repository verso un altro provider Git.
- AWS CodePipelineti aiuta a modellare e configurare rapidamente le diverse fasi di una versione del software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- Amazon Elastic Container Service (Amazon ECS) è un servizio rapido e scalabile di gestione dei container che ti aiuta a eseguire, arrestare e gestire container in un cluster.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- AWS Service Catalogconsente di gestire centralmente i cataloghi di servizi IT approvati. AWS Gli utenti finali possono distribuire rapidamente soltanto i servizi IT approvati di cui hanno bisogno, in accordo con i vincoli stabiliti dall'organizzazione.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri strumenti

- CloudFormation Linter (cfn-lint) è un linter che verifica i modelli CloudFormation YAML o JSON rispetto alle specifiche delle risorse. CloudFormation Esegue anche altri controlli, come la verifica di valori validi per le proprietà delle risorse e il rispetto delle migliori pratiche.
- cfn-nag è uno strumento open source che identifica potenziali problemi di sicurezza nei CloudFormation modelli mediante la ricerca di modelli.
- Docker è un insieme di prodotti Platform as a Service (PaaS) che utilizzano la virtualizzazione a livello di sistema operativo per fornire software in container. Questo modello utilizza Docker per creare e testare le immagini dei container localmente.
- Git è un sistema di controllo delle versioni distribuito e open source.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub dynamic_hotfix_codepipeline.

Best practice

Rivedi e modifica i ruoli IAM e le policy di controllo dei servizi (SCP) nel tuo ambiente per assicurarti che limitino l'accesso in modo appropriato. Questo è fondamentale per prevenire qualsiasi azione che possa prevalere sulle misure di sicurezza incluse in questo modello. Segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.

Epiche

Configura l'ambiente di lavoro

Operazione	Description	Competenze richieste
Clonare il repository.	Per clonare il repository di esempio in una nuova directory nella posizione di lavoro, esegui il comando seguente:	AWS DevOps

Operazione	Description	Competenze richieste
	`git clone git@github.com:aws-samples/dynamic_hotfix_codepipeline.git`	

Operazione	Description	Competenze richieste

Esporta le variabili di ambiente per la distribuzione CloudFormation in stack.	Definisci le seguenti variabili di ambiente che verranno utilizzate come input per gli CloudFormation stack più avanti in questo schema.
	• `ApplicationName` — Questa variabile viene utilizzata per denominare le risorse create per un'applicazione, semplificandone il tracciamento. Utilizzate il seguente comando, sostituendolo `Applicationname` con il nome effettivo dell'applicazione:
	`export ApplicationName=<Applicationname>`
	• `BucketStartName` — Questa variabile serve per denominare un bucket Amazon S3. I nomi dei bucket S3 devono essere univoci a livello globale per tutti. Account AWS Usa il seguente comando, sostituendolo `BucketName` con un nome univoco per il tuo bucket S3:
	`export BucketStartName=<BucketName>`
		AWS DevOps

Operazione	Description	Competenze richieste

	• Numeri di account e regioni: queste variabili memorizzano Account AWS i numeri per ambienti diversi e per la regione di distribuzione. Utilizza i seguenti comandi, sostituendo i segnaposto (come `prodaccountnumber eregion`) con i Account AWS numeri effettivi e quelli Regione AWS che stai utilizzando.
	
	Note
	`QAAccount` è facoltativo. Se desideri utilizzarlo`QAAccount`, configuralo utilizzando i parametri dello stack di pipeline principale.
	
	`export ProdAccount=<prodaccountnumber>`
	`export StageAccount=<stage/preprodaccountnumber>`
	`export QAAccount=<qaccountnumber>`
	`export ToolsAccount=<toolsaccountnumber>`	

Operazione	Description	Competenze richieste
	`export DepRegion =<region>`	

Imposta i prerequisiti richiesti in Account AWS

Operazione	Description	Competenze richieste
Crea le risorse necessarie e CI/CD nell'account degli strumenti.	Per distribuire lo CloudFormation stack nell'account degli strumenti, usa i seguenti comandi. (Rimuovi il `QAAccount` parametro se non utilizzi l'account QA per la configurazione.) `#InToolsAccount aws cloudformation deploy \ --template-file pre-requisites/pre-reqs.yaml \ --stack-name prereqs \ --parameter-overrides BucketStartName=${BucketStartName} \ ApplicationName=${ApplicationName} ProdAccount=${ProdAccount} \ StageAccount=${StageAccount} ToolsAccount=${ToolsAccount} \ QAAccount=${QAAccount} \ --capabilities CAPABILITY_IAM`	AWS DevOps

Operazione	Description	Competenze richieste
	`CAPABILITY_NAMED_IAM --region ${DepRegion}` Prendi nota delle risorse che il CodeCommit repository e Amazon ECR hanno creato dallo stack precedente. Questi parametri sono necessari per configurare il `main` ramo della pipeline nelle fasi successive.	

Operazione	Description	Competenze richieste

Crea le risorse necessarie per CI/CD gli account di carico di lavoro.	1. Per inserire il CloudFormation modello in ogni account di carico di lavoro (fase, produzione e controllo qualità opzionale), usa i seguenti comandi. Nel comando seguente, sostituiscilo `S3bucketpackage` con il nome del bucket Amazon S3 che desideri utilizzare per il packaging.
	
	`#InStageAccount`
	`aws cloudformation package \`
	`--template-file pre-requisites/infrastack.yaml \`
	`--s3-bucket <S3bucketpackage> \`
	`--s3-prefix infraStack \`
	`--region ${DepRegion} \`
	`--output-template-file pre-requisites/infrastructure_stage.template`
	
	`#InProdAccount`
	`aws cloudformation package \`
	`--template-file pre-requisites/infrastack.yaml \`
	`--s3-bucket <S3bucketpackage> \`
		AWS DevOps

Operazione	Description	Competenze richieste
	--s3-prefix infraStack \ --region ${DepRegion} \ --output-template-file pre-requisites/infrastructure_prod.template 2. Per distribuire il CloudFormation modello in ogni account di carico di lavoro, usa i seguenti comandi: #InStageAccount aws cloudformation deploy --stack-name inframainstack \ --parameter-overrides ApplicationName=${ApplicationName} ToolsAccount=${ToolsAccount} DepRegion=${DepRegion} \ --template-file pre-requisites/infrastructure_stage.template --region ${DepRegion} --capabilities CAPABILITY_NAMED_IAM #InProdAccount aws cloudformation deploy --stack-name inframainstack \ --parameter-overrides Applicati	

Operazione	Description	Competenze richieste
	onName=${Applicati onName} ToolsAcco unt=${ToolsAccount } DepRegion=${DepReg ion} \ --template-file pre-requisites/inf rastructure_prod.t emplate --region ${DepRegion} --capabilities CAPABILITY_NAMED_I AM	

Operazione	Description	Competenze richieste

Aggiorna la policy S3 Artifact Bucket per consentire l'accesso agli account dei carichi di lavoro.	Per aggiornare i prerequisiti dello CloudFormation stack nell'account tools, utilizza i seguenti comandi per aggiungere tutte le autorizzazioni necessarie per gli account di carico di lavoro Stage e Production. (Rimuovi il `QAAccount` parametro se non lo usi per la configurazione).
	
	`#InToolsAccount`
	`aws cloudformation deploy \`
	`--template-file pre-requisites/pre-reqs.yaml \`
	`--stack-name prereqs \`
	`--parameter-overrides BucketStartName=${BucketStartName} \`
	`ApplicationName=${ApplicationName} ProdAccount=${ProdAccount} \`
	`StageAccount=${StageAccount} ToolsAccount=${ToolsAccount} \`
	`QAAccount=${QAAccount} PutPolicy=true \`
	`--capabilities CAPABILITY_IAM CAPABILITY_NAMED_IAM --region ${DepRegion}`
		AWS DevOps

Configura la funzione Lambda e le risorse del Service Catalog nell'account degli strumenti

Operazione	Description	Competenze richieste

Configura il portafoglio e i prodotti Service Catalog.	Per configurare il portafoglio e i prodotti Service Catalog, procedi come segue:
	
	1. Carica i modelli pipeline-main.yaml e pipeline-hotfix.yaml dal repository nella CodePipeline directory in un bucket Amazon S3 esistente () nella regione in cui desideri effettuare la distribuzione (). `Bucketname` `DepRegion`
	
	`#InToolsAccount`
	`aws s3 cp ./codepipeline/pipeline-main.yaml s3://<Bucketname>/pipeline-main.yaml`
	`aws s3 cp ./codepipeline/pipeline-hotfix.yaml s3://<Bucketname>/pipeline-hotfix.yaml`
	
	2. Configura il portafoglio e il prodotto Service Catalog che gestiranno la pipeline per le `hotfix` filiali `main` e. Prendi nota dei dettagli contenuti nella `Outputs` sezione dedicata `MainProductId` e nello
		AWS DevOps

Operazione	Description	Competenze richieste
	`MainProductArtifactId` stack seguente. Le informazioni sono necessarie nelle fasi successive durante la configurazione della pipeline per la `main` filiale. `#InToolsAccount aws cloudformation deploy \ --template-file pre-requisites/servicecatalogsetup.yaml \ --stack-name servicecatalogsetup \ --parameter-overrides TemplateBucket=<Bucketname> \ --capabilities CAPABILITY_IAM CAPABILITY_NAMED_IAM --region ${DepRegion}` 3. Fornisci l'accesso a un ruolo IAM che distribuirà le risorse nell'account degli strumenti nel portafoglio di pipeline principale del portafoglio Service Catalog. Utilizza questo portafoglio per distribuire l'applicazione utilizzando la filiale. `main` Per ulteriori informazioni su come fornire l'accesso	

Operazione	Description	Competenze richieste
	, vedere Concessione dell'accesso agli utenti nella documentazione del Service Catalog.	

Operazione	Description	Competenze richieste
Configura le funzioni Lambda.	Questa soluzione utilizza le seguenti funzioni Lambda per gestire i flussi di lavoro degli hotfix: • `hotfix-lambda-function` gestisce il provisioning dei prodotti Service Catalog quando viene creata una `hotfix` filiale. • `hotfix-cleanup-lambda-function` gestisce la cessazione del prodotto quando viene eliminata una `hotfix` filiale. • `prcreation-lambda`crea richieste pull dal `hotfix` ramo ai `main` rami `develop` and. Per consentire alle funzioni Lambda di fornire e terminare i prodotti Service Catalog quando le `hotfix` filiali vengono create o eliminate tramite la EventBridge regola associata, utilizza i seguenti passaggi: 1. Per creare i ruoli e le autorizzazioni IAM per le funzioni Lambda, usa il seguente comando per	AWS DevOps

Operazione	Description	Competenze richieste
	distribuire CloudFormation uno stack: `#InToolsAccount aws cloudformation deploy \ --templat e-file pre-requi sites/lambdasetup. yaml \ --stack-n ame prsclambdasetup \ --capabilities CAPABILITY_IAM CAPABILITY_NAMED_I AM --region ${DepRegi on}` 2. Dopo la distribuzione dello stack, concedi l'`hotfix-la mbda-execution-rol e` accesso al portfolio di hotfix pipeline del portfolio Service Catalog utilizzando il. Console di gestione AWS Questo accesso consente alla funzione Lambda di avviare o terminare il prodotto Service Catalog per le filiali hotfix.	

Crea una pipeline per la filiale principale e distribuisci l'applicazione negli account dei carichi di lavoro

Operazione	Description	Competenze richieste
Configura la pipeline per la `main` filiale.	Per configurare la pipeline per il ramo principale, esegui il seguente comando nell'account degli strumenti	AWS DevOps

Operazione	Description	Competenze richieste
	. Sostituisci i parametri per `MainProductId` e `MainProductArtifactId` con i valori degli `servicecatalogsetup` stack output. `#InToolsAccount aws servicecatalog provision-product \ --product-id <MainProductId> \ --provisioning-artifact-id <MainProductArtifactId> \ --provisioned-product-name "${ApplicationName}-main-pipeline" \ --provisioning-parameters Key=CodeCommitRepoName,Value="${ApplicationName}-repository" Key=ECRRepository,Value="${ApplicationName}-app" \ --region=${DepRegion}`	

Operazione	Description	Competenze richieste
Distribuisci l'applicazione utilizzando la `main` filiale.	1. Per clonare il CodeCommit repository creato nelle prereq, usa il comando. `git clone` Per ulteriori informazioni, consulta Connect to the CodeCommit repository clonando il repository come descritto nella documentazione. CodeCommit 2. Copia tutti i file dell'applicazione dalla `repotemplates` directory disponibile nel repository nel clone del repository locale (). `${ApplicationName}-repository` Modificate i seguenti file per aggiornare l'ID. `ToolsAccount` In ogni file, individuate il `RegistryAccountid` parametro e impostatene il valore sul vostro `ToolsAccount` ID. Esegui il commit delle modifiche nel CodeCommit repository e invia i file ai `develop` rami `main` e. • ecs-configuration-prod.yaml • ecs-configuration-qa.yaml • ecs-configuration-stage.yaml	AWS DevOps

Operazione	Description	Competenze richieste
	3. Per verificare la distribuzione dell'applicazione, monitorate l' CodePipeline esecuzione utilizzando. Console di gestione AWS Una volta completata la distribuzione, accedi all'applicazione utilizzando l'FQDN di Application Load Balancer nell'ambiente stage. Verificate che l'applicazione funzioni come previsto. 4. Per approvare la distribuzione in produzione, utilizza la CodePipeline console per individuare la pipeline per l'applicazione. Trova il palco. `ApprovalToStart` Esamina le modifiche e, se soddisfacenti, fornisci l'approvazione manuale per procedere con l'implementazione in produzione.	

Crea la pipeline per un ramo hotfix-* e distribuisci l'hotfix

Operazione	Description	Competenze richieste
Crea una `hotfix-*` filiale e applica le modifiche.	Per creare una pipeline per la `hotfix-*` filiale e distribuire l'hotfix negli account del carico di lavoro, procedi come segue:	AWS DevOps

Operazione	Description	Competenze richieste
	1. Crea un ramo utilizzando un nome che inizia con la parola chiave. `hotfix` Ad esempio, questo modello utilizza il `hotfix-check1` ramo nel repository dell' CodeCommit applicazione (`${ApplicationName}-repository`). Per passaggi più dettaglia ti, consulta Connect to an AWS CodeCommit repository e i comandi Git di base nella CodeCommit documentazione. 2. Verificare che il prodotto Service Catalog `Hotfix CICD Pipeline` sia stato fornito correttam ente in modo dinamico per la `hotfix-check1` filiale. Il nome del prodotto fornito prende il nome dal nome del ramo dell'hotfix e dal nome del repositor y dell'applicazione. CodeCommit 3. Effettua alcune modifiche minori nel file index.html e inseriscile nel repository. CodeCommit 4. Verificate che l' CodePipel ine esecuzione abbia esito positivo nell'ambiente di	

Operazione	Description	Competenze richieste
	stage. Per la distribuzione nell'ambiente di produzione, fornisci l'approvazione manuale in CodePipeline. 5. Verifica che le modifiche siano visibili nella home page dell'applicazione utilizzando il nome di dominio completo (FQDN) di Application Load Balancer. L'FQDN è disponibile nella sezione di. `Outputs` `inframain stack-ALBStack-*`	
Elimina il `hotfix-check1` ramo.	Per eliminare il `hotfix-check1` ramo creato in precedenza, procedi come segue: 1. Eliminate il `hotfix-check1` ramo che si trova nel repository CodeCommit dell'applicazione. 2. Verificare che il prodotto Service Catalog fornito per la `hotfix-check1` filiale sia stato terminato correttamente.	AWS DevOps

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste

Pulisci le risorse distribuite.	Per ripulire le risorse distribuite in precedenza, procedi come segue:
	1. Per ridimensionare il servizio Amazon ECS fino a zero repliche negli account di carico di lavoro, usa il seguente comando:
	`aws ecs update-service --cluster ${ApplicationName}-Cluster --service ${ApplicationName}-Service-stage --desired-count 0 --region ${DepRegion}`
	`aws ecs update-service --cluster ${ApplicationName}-Cluster --service ${ApplicationName}-Service-prod --desired-count 0 --region ${DepRegion}`
	2. Terminare il prodotto Service Catalog fornito per la `main` filiale.
	3. Pulisci gli oggetti creati nei bucket Amazon S3 nell'account degli strumenti . Elimina tutte le immagini Docker in Amazon ECR
		AWS DevOps

Operazione	Description	Competenze richieste

	prima di eliminare il registro stesso.
	4. Rimuovi i ruoli IAM nella sezione di accesso concesso dei portafogl i Service Catalog prima di eliminare il portafoglio Service Catalog.
	5. Elimina gli CloudFormation stack distribuiti nell'acco unt degli strumenti e negli account del carico di lavoro.
	
	`##In Tools Account##`
	`aws cloudformation delete-stack --stack-name servicecatalogsetup --region ${DepRegion}`
	`aws cloudformation delete-stack --stack-name prlambdasetup --region ${DepRegion}`
	`aws cloudformation delete-stack --stack-name prereqs --region ${DepRegion}`
	
	`##In Workload Accounts##`
	`aws cloudformation delete-stack --stack-name inframainstack --region ${DepRegion}`	

Operazione	Description	Competenze richieste
	Per ulteriori informazioni, vedere Eliminazione dei prodotti forniti nella documentazione del Service Catalog.	

Risoluzione dei problemi

Problema	Soluzione
Le modifiche che hai apportato al CodeCommit repository non vengono distribuite.	Controlla la presenza di errori nei CodeBuild log nell'azione di compilazione di Docker. Per ulteriori informazioni, consulta la documentazione relativa ad CodeBuild .
Il prodotto Service Catalog non viene fornito.	Esamina gli CloudFormation stack correlati per verificare la presenza di eventi non riusciti. Per ulteriori informazioni, consulta la documentazione relativa ad CloudFormation .

Risorse correlate

- Comandi Git di base
- Configura una policy IAM per limitare i push e le unioni a una filiale
- Connect a un AWS CodeCommit repository
- Concessione dell'accesso agli utenti
- Trasferimento di un'immagine Docker a un repository privato Amazon ECR
- Risoluzione dei problemi AWS CodeBuild
- AWS CodePipeline Che cos'è?

Informazioni aggiuntive

Questo modello è progettato per ambienti con una configurazione Gitflow adottata per il flusso di lavoro di sviluppo del CI/CD processo. Le pipeline seguono il ciclo di implementazione che parte dallo

sviluppo e passa attraverso il controllo della qualità (QA), la fase e gli ambienti di produzione. La CI/CD configurazione include due filiali git con implementazioni promozionali negli ambienti come segue:

- La `develop` filiale si distribuisce nell'ambiente di sviluppo.
- La `main` filiale si occupa degli ambienti di controllo qualità, stage e produzione.

In questa configurazione, è difficile applicare un hotfix o una patch di sicurezza più velocemente del normale ciclo di distribuzione mentre è in corso lo sviluppo attivo di nuove funzionalità. È necessario un processo dedicato per rispondere agli hotfix o alle richieste di sicurezza, garantendo che gli ambienti live rimangano correttamente funzionanti e sicuri.

Tuttavia, è possibile utilizzare altre opzioni disponibili senza richiedere un processo di distribuzione dedicato se:

- Il CI/CD processo è ben equipaggiato con test automatizzati, come test funzionali e end-to-end test, che eliminano la necessità di test manuali e impediscono ritardi nelle implementazioni in produzione. Tuttavia, se i test automatici non sono ben integrati nel CI/CD processo, apportare una piccola correzione all'ambiente di produzione può diventare complesso e complicato per gli sviluppatori. Questo perché potrebbero esserci nuove funzionalità in attesa di approvazione e approvazione nell'ambiente di controllo qualità. Un hotfix o una correzione di sicurezza non possono essere messi in produzione in modo semplice e simultaneo.
- I team di sviluppo implementano continuamente nuove funzionalità nell'ambiente di produzione, integrando hotfix o patch di sicurezza nella distribuzione pianificata di ogni nuova funzionalità. In altre parole, il prossimo aggiornamento delle funzionalità dell'ambiente di produzione comprende due componenti: l'aggiunta di una nuova funzionalità e l'inclusione dell'hotfix o della patch di sicurezza. Tuttavia, se il ciclo di implementazione non è continuo, possono esserci più nuove funzionalità in attesa di approvazione nell'ambiente di controllo qualità. La gestione di versioni diverse e la garanzia che vengano riapplicate le modifiche corrette possono quindi diventare complesse e soggette a errori.

> **Note**
>
> Se utilizzi la versione 2 di AWS CodePipeline con i trigger appropriati configurati nella `hotfix` filiale, avrai comunque bisogno di un processo dedicato per rispondere alle richieste non pianificate. Nella versione 2, puoi impostare i trigger per le richieste push o pull. L'esecuzione verrà messa in coda o eseguita immediatamente, a seconda dello stato precedente della pipeline. Tuttavia, con una pipeline dedicata, le correzioni vengono applicate

immediatamente all'ambiente di produzione, garantendo che i problemi urgenti vengano risolti senza indugio.

Eliminazione automatica degli AWS CloudFormation stack e delle risorse associate

SANDEEP SINGH e James Jacob, Amazon Web Services

Riepilogo

AWS CloudFormationè un servizio ampiamente utilizzato per la gestione dell'infrastruttura cloud come codice (IaC). Quando lo utilizzi CloudFormation, gestisci le risorse correlate come una singola unità chiamata stack. Puoi creare, aggiornare ed eliminare una raccolta di risorse mediante la creazione, l'aggiornamento e l'eliminazione di stack.

A volte, non sono più necessarie le risorse in una CloudFormation pila. A seconda delle risorse e delle relative configurazioni, può essere complicato eliminare uno stack e le risorse associate. Nei sistemi di produzione reali, le eliminazioni a volte falliscono o richiedono molto tempo a causa di condizioni o restrizioni contrastanti che non possono essere annullate. CloudFormation Può richiedere un'attenta pianificazione ed esecuzione per garantire che tutte le risorse vengano eliminate correttamente in modo efficiente e coerente. Questo modello descrive come configurare un framework che consenta di gestire l'eliminazione di CloudFormation stack che comportano le seguenti complessità:

- Risorse con protezione da eliminazione: per alcune risorse potrebbe essere abilitata la protezione da eliminazione. Esempi comuni sono le tabelle Amazon DynamoDB e i bucket Amazon Simple Storage Service (Amazon S3). La protezione da eliminazione impedisce l'eliminazione automatica, ad esempio l'eliminazione totale. CloudFormation Se si desidera eliminare queste risorse, è necessario ignorare o disabilitare temporaneamente la protezione da eliminazione manualmente o programmaticamente. È necessario considerare attentamente le implicazioni dell'eliminazione di queste risorse prima di procedere.
- Risorse con politiche di conservazione: alcune risorse, come AWS Key Management Service (AWS KMS) chiavi () e bucket Amazon S3, potrebbero avere politiche di conservazione che specificano per quanto tempo devono essere conservate dopo la richiesta di eliminazione. È necessario tenere conto di queste politiche nella strategia di pulizia per mantenere la conformità alle politiche organizzative e ai requisiti normativi.
- Eliminazione ritardata delle funzioni Lambda collegate a un VPC: l'eliminazione di una funzione collegata a AWS Lambdaun cloud privato virtuale (VPC) può richiedere dai 5 ai 40 minuti, a seconda delle molteplici dipendenze interconnesse coinvolte nel processo. Se scolleghi la funzione dal VPC prima di eliminare lo stack, puoi ridurre questo ritardo a meno di 1 minuto.

- Risorse non create direttamente da CloudFormation: in alcuni progetti di applicazioni, le risorse potrebbero essere create all'esterno dello CloudFormation stack originale, dall'applicazione stessa o dalle risorse fornite tramite lo stack. Di seguito sono riportati due esempi:
  - CloudFormation potrebbe fornire un'istanza Amazon Elastic Compute Cloud (Amazon EC2) che esegue uno script di dati utente. Quindi, questo script potrebbe creare un AWS Systems Managerparametro per archiviare i dati relativi all'applicazione. Questo parametro non è gestito tramite. CloudFormation
  - CloudFormation potrebbe fornire una funzione Lambda che genera automaticamente un gruppo Amazon CloudWatch Logs per l'archiviazione dei log. Questo gruppo di log non è gestito tramite. CloudFormation

  Anche se queste risorse non sono gestite direttamente da CloudFormation, spesso devono essere ripulite quando lo stack viene eliminato. Se non vengono gestite, possono diventare orfane e portare a un consumo inutile di risorse.

Sebbene queste barriere possano causare complessità, sono intenzionali e critiche. Consentire CloudFormation di ignorare tutti i vincoli e di eliminare indiscriminatamente le risorse potrebbe portare a conseguenze dannose e impreviste in molti scenari. Tuttavia, in qualità di ingegnere del cloud responsabile della gestione dell'ambiente, a volte potrebbe essere necessario ignorare questi vincoli, in particolare negli ambienti di sviluppo, test DevOps o gestione temporanea.

Risultati aziendali mirati

Implementando questo framework, è possibile ottenere i seguenti vantaggi:

- Gestione dei costi: la pulizia regolare ed efficiente degli ambienti temporanei, ad esempio degli ambienti di test di accettazione da parte degli utenti, aiuta a evitare che le risorse funzionino più a lungo del necessario. end-to-end Ciò può ridurre significativamente i costi.
- Sicurezza: la pulizia automatizzata delle risorse obsolete o inutilizzate riduce la superficie di attacco e aiuta a mantenere un ambiente sicuro. AWS
- Efficienza operativa: una pulizia regolare e automatizzata può offrire i seguenti vantaggi operativi:
  - Gli script automatici che rimuovono vecchi gruppi di log o bucket Amazon S3 vuoti possono migliorare l'efficienza operativa mantenendo l'ambiente pulito e gestibile.
  - L'eliminazione e la ricreazione rapide degli stack supportano un'iterazione rapida per la progettazione e l'implementazione, che può portare a un'architettura più solida e resiliente.

- L'eliminazione e la ricostruzione regolari degli ambienti possono aiutarti a identificare e risolvere potenziali problemi. Questo può aiutarti a garantire che l'infrastruttura sia in grado di resistere a scenari del mondo reale.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Python versione 3.6 o successiva, installata
- AWS Command Line Interface (AWS CLI), installato e configurato

Limitazioni

- Viene utilizzata una convenzione di denominazione per identificare le risorse da eliminare. Il codice di esempio in questo modello utilizza un prefisso per il nome della risorsa, ma è possibile definire una convenzione di denominazione personalizzata. Le risorse che non utilizzano questa convenzione di denominazione non verranno identificate o successivamente eliminate.

Architecture

Il diagramma seguente mostra come questo framework identifica lo CloudFormation stack di destinazione e le risorse aggiuntive ad esso associate.

Il diagramma mostra il flusso di lavoro seguente:

1. Raccogliere risorse: il framework di automazione utilizza una convenzione di denominazione per restituire tutti gli CloudFormation stack pertinenti, i repository Amazon Elastic Container Registry (Amazon ECR), le tabelle DynamoDB e i bucket Amazon S3.

> **Note**
>
> Le funzioni per questa fase utilizzano i paginatori, una funzionalità di Boto3 che astrae il processo di iterazione su un set di risultati API troncato. Questo assicura che tutte le risorse vengano elaborate. Per ottimizzare ulteriormente le prestazioni, prendete in

considerazione l'applicazione del filtro lato server o l'utilizzo JMESPath per eseguire il filtraggio lato client.

2. Preelaborazione: il framework di automazione identifica e risolve i vincoli del servizio che devono essere ignorati per consentire l'eliminazione delle risorse. CloudFormation Ad esempio, modifica l'`DeletionProtectionEnabled`impostazione per le tabelle DynamoDB in. `False` Nell'interfaccia a riga di comando, per ogni risorsa, viene visualizzato un messaggio che chiede se si desidera sovrascrivere il vincolo.
3. Elimina stack: il framework di automazione elimina lo stack. CloudFormation Nell'interfaccia a riga di comando, viene visualizzato un messaggio che richiede se si desidera eliminare lo stack.
4. Postelaborazione: il framework di automazione elimina tutte le risorse correlate che non sono state fornite direttamente come parte dello stack. CloudFormation Esempi di questi tipi di risorse includono i parametri e i gruppi di CloudWatch log di Systems Manager. Funzioni separate raccolgono queste risorse, le preelaborano e quindi le eliminano. Nell'interfaccia a riga di comando, per ogni risorsa, viene visualizzato un messaggio che chiede se si desidera eliminare la risorsa.

> **Note**
>
> Le funzioni per questa fase utilizzano gli impaginatori, una funzionalità di Boto3 che astrae il processo di iterazione su un set di risultati API troncato. Questo assicura che tutte le risorse vengano elaborate. Per ottimizzare ulteriormente le prestazioni, prendete in considerazione l'applicazione del filtro lato server o l'utilizzo JMESPath per eseguire il filtraggio lato client.

Automazione e scalabilità

Se CloudFormation lo stack include altre risorse che non sono incluse nel codice di esempio o se lo stack presenta un vincolo che non è stato risolto in questo schema, puoi adattare il framework di automazione al tuo caso d'uso. Segui la stessa metodologia di raccolta delle risorse, pre-elaborazione, eliminazione dello stack e quindi post-elaborazione.

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita in e. Account AWS Regioni AWS
- CloudFormation Command Line Interface (CFN-CLI) è uno strumento open source che ti aiuta a sviluppare e testare estensioni di terze parti AWS e quindi a registrarle per l'uso in. CloudFormation
- AWS SDK per Python (Boto3)è un kit di sviluppo software che ti aiuta a integrare la tua applicazione, libreria o script Python con. Servizi AWS

Altri strumenti

- Click è uno strumento Python che consente di creare interfacce a riga di comando.
- Poetry è uno strumento per la gestione delle dipendenze e la creazione di pacchetti in Python.
- Pyenv è uno strumento che ti aiuta a gestire e passare da una versione di Python all'altra.
- Python è un linguaggio di programmazione per computer generico.

Deposito di codice

Il codice per questo pattern è disponibile nel GitHub cloudformation-stack-cleanuprepository.

Best practice

- Etichettate le risorse per una facile identificazione: implementate una strategia di etichettatura per identificare le risorse create per ambienti e scopi diversi. I tag possono semplificare il processo di pulizia aiutandoti a filtrare le risorse in base ai relativi tag.
- Imposta i cicli di vita delle risorse: definisci i cicli di vita delle risorse per eliminarle automaticamente dopo un determinato periodo. Questa pratica consente di garantire che gli ambienti temporanei non diventino oneri di costo permanenti.

Epiche

Installa strumenti

Operazione	Description	Competenze richieste
Clonare il repository.	1. Crea una cartella nel tuo ambiente virtuale. Assegna	DevOps ingegnere

Operazione	Description	Competenze richieste
	un nome con il nome del tuo progetto. 2. Apri un terminale sul tuo computer locale e accedi a questa cartella. 3. Inserisci il seguente comando per clonare il cloudformation-stack-cleanup repository nella directory del tuo progetto: `git clone https://github.com/aws-samples/cloudformation-stack-cleanup.git`	
Installa Poetry.	Segui le istruzioni (documentazione di Poetry) per installare Poetry nell'ambiente virtuale di destinazione.	DevOps ingegnere

Operazione	Description	Competenze richieste

Installare le dipendenze.	1. Immettete il seguente comando per accedere alla directory del progetto:
	`cd cloudformation-stack-cleanup`
	2. Immetti il comando seguente:
	`poetry install`
	Questo installa tutte le dipendenze richieste , come Boto3, click e il codice sorgente per la CLI. CloudFormation
		DevOps ingegnere

(Opzionale) Installa Pyenv.	Segui le istruzioni (GitHub) per installare Pyenv.	DevOps ingegnere

(Facoltativo) Personalizza il framework

Operazione	Description	Competenze richieste
Crea funzioni che raccolgono, preelaborano ed eliminano le risorse di destinazione.	1. Nel repository clonato, immettete il seguente comando per accedere alla directory: `cli` `cd cfncli/cli` 2. Aprire il file cleanup_environment.py.	DevOps ingegnere, Python

Operazione	Description	Competenze richieste
	3. Crea una nuova funzione Python che raccolga il tipo di risorsa che desideri modificare. Per un esempio, vedete la `gather_ddb_tables` funzione in questo file. 4. Crea una nuova funzione Python che sovrascriva i vincoli di servizio per la risorsa di destinazione. Per un esempio, vedete la funzione in questo file. `remove_ddb_deletion_protection` 5. Crea una nuova funzione Python che raccolga risorse di destinazione non gestite. Per un esempio, vedete la `gather_log_groups` funzione in questo file. 6. Crea una nuova funzione Python che elimini le risorse di destinazione non gestite. Per un esempio, vedete la `delete_log_group` funzione in questo file. 7. Salvate e chiudete il file cleanup_enviornment.py.	

Crea risorse di esempio

Operazione	Description	Competenze richieste
Crea una CloudFormation pila.	1. Passa alla directory del progetto . 2. Immettete il comando seguente per creare uno CloudFormation stack che esegua il provisioning di una tabella DynamoDB e di un gruppo di sicurezza . Aggiorna il valore per: `<VPCID>` `aws cloudformation create-stack \ --stack-name sampleforcleanup-Stack \ --template-body file://samples/sample-cfn-stack.yaml \ --parameters ParameterKey=VpcId,ParameterValue=<VPCID> \ --region us-east-1`	AWS DevOps
Creare un parametro Systems Manager.	Immettete il comando seguente per creare un parametro Systems Manager il cui provisioning non viene eseguito tramite CloudFormation: `aws ssm put-parameter \`	AWS DevOps

Operazione	Description	Competenze richieste
	`--name "/samplef orcleanup/database/ password" \ --value "your_db_ password" \ --type "SecureString" \ --description "Database password for my app" \ --tier "Standard" \ --region "us-east-1"`	
Creare un bucket Amazon S3.	Immetti il seguente comando per creare un bucket Amazon S3 che non viene fornito tramite: CloudFormation `aws s3api create-bucket \ --bucket samplesor cleanup-unmanagedb ucket-<UniqueIdent ifier> \ --region us-east-1 \ --create-bucket-co nfiguration LocationC onstraint=us-east-1`	AWS DevOps

Elimina le risorse di esempio

Operazione	Description	Competenze richieste
Elimina lo CloudFormation stack.	1. Immetti il seguente comando per eliminare lo CloudFormation stack di esempio, il parametro	AWS DevOps

Operazione	Description	Competenze richieste
	Systems Manager e il bucket Amazon S3 che hai creato: `cfncli --region us-east-1 \ dev cleanup-env \ --prefix-list sampleforcleanup` 2. Quando richiesto, inserisci per continuare. Y	
Convalida l'eliminazione delle risorse.	Nell'output, verifica che tutte le risorse di esempio siano state eliminate. Per un esempio di output, consultate la sezione Risorse aggiuntive di questo modello.	AWS DevOps

Risorse correlate

- Eliminare uno stack (CloudFormation documentazione)
- Risoluzione dei problemi CloudFormation (CloudFormation documentazione)
- Fornire alle funzioni Lambda l'accesso alle risorse in un Amazon VPC (documentazione Lambda)
- Come posso eliminare uno AWS CloudFormation stack bloccato nello stato DELETE_FAILED? (Centro di conoscenza)AWS

Informazioni aggiuntive

Di seguito è riportato un esempio di output del `cfncli` comando:

```
cfncli --region aus-east-1  dev cleanup-env --prefix-list sampleforcleanup


https://sts.us-east-1.amazonaws.com
```

```
Cleaning up: ['sampleforcleanup'] in xxxxxxxxxx:us-east-1
Do you want to proceed? [Y/n]: Y
No S3 buckets
No ECR repositories
No Lambda functions in VPC
The following DynamoDB tables will have their deletion protection removed:
sampleforcleanup-MyDynamoDBTable
Do you want to proceed with removing deletion protection from these tables? [Y/n]: Y
Deletion protection disabled for DynamoDB table 'sampleforcleanup-MyDynamoDBTable'.
The following CloudFormation stacks will be deleted:
sampleforcleanup-Stack
Do you want to proceed with deleting these CloudFormation stacks? [Y/n]: Y
Initiated deletion of CloudFormation stack: `sampleforcleanup-Stack`
Waiting for stack `sampleforcleanup-Stack` to be deleted...
CloudFormation stack `sampleforcleanup-Stack` deleted successfully.
The following ssm_params will be deleted:
/sampleforcleanup/database/password
Do you want to proceed with deleting these ssm_params? [Y/n]: Y
Deleted SSM Parameter: /sampleforcleanup/database/password
Cleaned up: ['sampleforcleanup']
```

Automatizza l'inserimento e la visualizzazione di metriche personalizzate di Amazon MWAA su Amazon Managed Grafana utilizzando Terraform

Faisal Abdullah e Satya Vajrapu, Amazon Web Services

Riepilogo

Questo modello illustra come utilizzare Amazon Managed Grafana per creare e monitorare metriche personalizzate che vengono acquisite da Amazon Managed Workflows for Apache Airflow (Amazon MWAA). Amazon MWAA funge da orchestratore per i flussi di lavoro, utilizzando Directed Acyclic Graphs () con script in Python. DAGs Questo modello è incentrato sul monitoraggio di metriche personalizzate, tra cui il numero totale di esecuzioni nell'ultima ora, il conteggio dei DAGs risultati superati e non riusciti ogni ora e la durata media di questi processi. DAGs Questa analisi mostra come Amazon Managed Grafana si integra con Amazon MWAA per consentire un monitoraggio completo e approfondimenti sull'orchestrazione dei flussi di lavoro all'interno di questo ambiente.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS con le autorizzazioni utente necessarie per creare e gestire quanto segue: Servizi AWS
  - AWS Identity and Access Management Ruoli e politiche (IAM)
  - AWS Lambda
  - Grafana gestito da Amazon
  - Amazon Managed Workflows for Apache Airflow (Amazon MWAA)
  - Amazon Simple Storage Service (Amazon S3)
  - Amazon Timestream
- Accesso a un ambiente shell che può essere un terminale sul computer locale o AWS CloudShell.
- Un ambiente shell con Git installato e l'ultima versione di AWS Command Line Interface (AWS CLI) installata e configurata. Per ulteriori informazioni, consulta Installazione o aggiornamento alla versione più recente di AWS CLI nella AWS CLI documentazione.
- È installata la seguente versione di Terraform: `required_version = ">= 1.6.1, < 2.0.0"` Puoi usare tfswitch per passare da una versione all'altra di Terraform.
- Fonte di identità configurata per il tuo. AWS IAM Identity Center Account AWS Per ulteriori informazioni, consulta Conferma le tue fonti di identità in IAM Identity Center nella documentazione

di IAM Identity Center. Puoi scegliere tra l'impostazione predefinita IAM Identity Center directory, Active Directory o un provider di identità (IdP) esterno come Okta. Per ulteriori informazioni, vedere Risorse correlate.

Limitazioni

- Alcune Servizi AWS non sono disponibili in tutte Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- Terraform `required_version = ">= 1.6.1, < 2.0.0"`
- Amazon Managed Grafana versione 9.4 o successiva. Questo modello è stato testato nella versione 9.4.

Architecture

Il seguente diagramma di architettura evidenzia l' Servizi AWS uso nella soluzione.

Il diagramma precedente illustra il seguente flusso di lavoro:

1. Le metriche personalizzate all'interno di Amazon MWAA provengono da DAGs quelle eseguite all'interno dell'ambiente. Le metriche vengono caricate nel bucket Amazon S3 in un formato di file CSV. Di seguito vengono DAGs utilizzate le funzionalità di interrogazione del database di Amazon MWAA:
   - `run-example-dag`— Questo DAG contiene codice Python di esempio che definisce una o più attività. Viene eseguito ogni 7 minuti e stampa la data. Dopo aver stampato la data, il DAG include un'operazione per sospendere o sospendere l'esecuzione per una durata specifica.
   - `other-sample-dag`— Questo DAG viene eseguito ogni 10 minuti e stampa la data. Dopo aver stampato la data, il DAG include un'operazione per sospendere o sospendere l'esecuzione per una durata specifica.
   - `data-extract`— Questo DAG viene eseguito ogni ora, interroga il database Amazon MWAA e raccoglie i parametri. Dopo aver raccolto i parametri, questo DAG li scrive in un bucket Amazon S3 per ulteriori elaborazioni e analisi.

2. Per semplificare l'elaborazione dei dati, le funzioni Lambda vengono eseguite quando vengono attivate dagli eventi di Amazon S3, il che facilita il caricamento delle metriche in Timestream.
3. Timestream è integrato come fonte di dati in Amazon Managed Grafana, dove sono archiviate tutte le metriche personalizzate di Amazon MWAA.
4. Gli utenti possono interrogare i dati e creare dashboard personalizzati per visualizzare gli indicatori chiave delle prestazioni e ottenere informazioni sull'orchestrazione dei flussi di lavoro all'interno di Amazon MWAA.

Tools (Strumenti)

Servizi AWS

- AWS IAM Identity Centerti aiuta a gestire centralmente l'accesso Single Sign-On (SSO) a tutte le tue applicazioni e al cloud. Account AWS
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi. In questo modello, AWS Lambda esegue il codice Python in risposta agli eventi di Amazon S3 e gestisce automaticamente le risorse di calcolo.
- Amazon Managed Grafana è un servizio di visualizzazione dei dati completamente gestito che puoi utilizzare per interrogare, correlare, visualizzare e inviare avvisi su metriche, log e tracce. Questo modello utilizza Amazon Managed Grafana per creare una dashboard per la visualizzazione delle metriche e gli avvisi.
- Amazon Managed Workflows for Apache Airflow (Amazon MWAA) è un servizio di orchestrazione gestito per Apache Airflow che puoi utilizzare per configurare e gestire pipeline di dati nel cloud su larga scala. Apache Airflow è uno strumento open source utilizzato per creare, pianificare e monitorare in modo programmatico sequenze di processi e attività denominate flussi di lavoro. In questo modello, in Amazon MWAA vengono distribuiti un campione DAGs e un estrattore di parametri DAG.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati. In questo modello, Amazon S3 viene utilizzato per DAGs archiviare script e metriche personalizzate in formato CSV.
- Amazon Timestream LiveAnalytics for is è un database di serie temporali veloce, scalabile, completamente gestito e creato appositamente che semplifica l'archiviazione e l'analisi di trilioni di

punti dati di serie temporali al giorno. Timestream for si integra LiveAnalytics anche con i servizi di uso comune per la raccolta, la visualizzazione e l'apprendimento automatico dei dati. In questo modello, viene utilizzato per importare i parametri personalizzati Amazon MWAA generati.

Altri strumenti

- HashiCorp Terraform è uno strumento Infrastructure as Code (IaC) che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud. Questo modello utilizza un modulo Terraform per automatizzare la fornitura dell'infrastruttura in. AWS

Archivio di codice

Il codice per questo pattern è disponibile GitHub nel repository visualize-amazon-mwaa-custom-metrics-grafana. La cartella contiene quanto segue: `stacks/Infra`

- File di configurazione Terraform per tutte le risorse AWS
- File Grafana dashboard .json nella cartella `grafana`
- Amazon Managed Workflows for Apache DAGs Airflow nella cartella `mwaa/dags`
- Codice Lambda per analizzare il file.csv e memorizzare le metriche nel database Timestream nella cartella `src`
- I file IAM Policy .json nella cartella `templates`

Best practice

Terraform deve memorizzare lo stato dell'infrastruttura e della configurazione gestite in modo da poter mappare le risorse del mondo reale alla configurazione. Per impostazione predefinita, Terraform memorizza lo stato localmente in un file denominato. `terraform.tfstate` È fondamentale garantire la sicurezza e l'integrità del file di stato Terraform perché mantiene lo stato attuale dell'infrastruttura. Per ulteriori informazioni, consulta Remote State nella documentazione di Terraform.

Epiche

Implementa l'infrastruttura utilizzando Terraform

Operazione	Description	Competenze richieste

Implementa l'infrastruttura.	Per implementare l'infrast ruttura della soluzione, procedi come segue:
	1. Apri un terminale o un prompt dei comandi sul tuo computer locale o utilizzan do. AWS CloudShell
	2. Passa alla directory in cui desideri clonare il repositor y.
	3. Per clonare il repositor y, esegui il seguente comando:
	`git clone https://g ithub.com/aws-samp les/visualize-amaz on-mwaa-custom-met rics-grafana`
	4. Al termine del processo di clonazione, esegui il comando seguente per accedere alla directory del repository clonato:
	`cd visualize-amazon-m waa-custom-metrics- grafana/stacks/infra`
		AWS DevOps

Operazione	Description	Competenze richieste

	5. Per scaricare e inizializzare i provider richiesti, esegui il comando seguente:
	`terraform init`
	6. Per avere una visione completa di tutte le risorse che Terraform creerà, esegui il seguente comando:
	`terraform plan`
	Terraform fornisce le seguenti risorse:
	• Amazon Virtual Private Cloud (Amazon VPC) e componenti di rete associati
	• Risorse di Amazon S3
	• AWS Lambda funzioni
	• Risorse Amazon Managed Grafana (area di lavoro, dashboard, origine dati)
	• Supporto delle risorse IAM (ruoli e politiche)
	7. Per creare le AWS risorse dall'output del piano, esegui il comando seguente:
	`terraform apply -auto-approve`	

Operazione	Description	Competenze richieste
	Il provisioning dell'infr astruttura viene completato in circa 20 minuti. 8. Per creare le AWS risorse specificate in base alla configurazione definita nei file Terraform, esegui il seguente comando: `terraform apply`	

Convalida le risorse dell'infrastruttura distribuite

Operazione	Description	Competenze richieste
Convalida l'ambiente Amazon MWAA.	Per convalidare l'ambiente Amazon MWAA, procedi come segue: 1. Accedi a Console di gestione AWS, accedi alla console dashboard di Amazon MWAA e seleziona Open Airflow UI. 2. Dovresti vedere i seguenti tre DAGs in stato Attivo: • estratto di dati • run-example-dag • other-sample-dag 3. Se un DAG non è attivo, puoi attivarlo attivando	AWS DevOps, ingegnere dei dati

Operazione	Description	Competenze richieste
	l'interruttore a levetta accanto al nome del DAG.	
Verifica gli orari del DAG.	Per visualizzare ogni pianificazione DAG, vai alla scheda Pianificazione nell'interfaccia utente Airflow. Ciascuno dei seguenti sistemi DAGs ha una pianificazione preconfigurata, che viene eseguita nell'ambiente Amazon MWAA e genera parametri personalizzati: • run-example-dag- Viene eseguito ogni 7 minuti • other-sample-dag- Funziona ogni 10 minuti • data-extract: viene eseguito ogni ora Puoi anche vedere le esecuzioni riuscite di ogni DAG nella colonna Runs.	Ingegnere dei dati, AWS DevOps

Configura l'ambiente Amazon Managed Grafana

Operazione	Description	Competenze richieste
Configura l'accesso all'area di lavoro Amazon Managed Grafana.	Gli script Terraform hanno creato lo spazio di lavoro, i dashboard e la pagina delle metriche di Amazon	AWS DevOps

Operazione	Description	Competenze richieste
	Managed Grafana richiesti. Per configurare l'accesso in modo da poterli visualizzare, procedi come segue: 1. Apri la console Amazon Managed Grafana. 2. In Workspaces, seleziona l'area di lavoro `grafana-ws-dev` e vai alla scheda Autenticazione nel riquadro inferiore. 3. Scegli il pulsante Assegna nuovo utente o gruppo. 4. Aggiungi il tuo gruppo nella scheda Gruppi o un utente nella scheda Utenti, quindi scegli il pulsante Assegna utente e gruppi. 5. Dopo aver aggiunto l'utente (o il gruppo), imposta questo utente (o gruppo) come amministratore. Seleziona l'utente in Utenti o gruppo assegnati nella scheda Gruppo di utenti assegnati e scegli Rendi amministratore dal menu a discesa. Per ulteriori informazioni, consulta Use AWS IAM Identity Center with your Amazon Managed Grafana workspace nella	

Operazione	Description	Competenze richieste
	documentazione di Amazon Managed Grafana. 6. Passa a Workspaces, quindi scegli l'URL dell'area di lavoro Grafana. Per accedere ad Amazon Managed Grafana come amministratore, scegli Accedi con. AWS IAM Identity Center	

Operazione	Description	Competenze richieste
Installa il plug-in Amazon Timestream.	Le metriche personalizzate di Amazon MWAA vengono caricate nel database Timestream. Utilizza il plug-in Timestream per visualizzare le metriche con i dashboard di Amazon Managed Grafana. Per installare il plug-in Timestream, procedi come segue: 1. Nella console Amazon Managed Grafana, espandi il menu nel riquadro di navigazione a sinistra e vai a Amministrazione, Plugin. 2. Cerca e installa la versione più recente del plug-in Amazon Timestream. 3. Dopo aver installato il plug-in, vai su Amministrazione, Fonti dati per vedere la fonte di dati Timestream. Se l'origine dati non è elencata, aggiorna la pagina. Per ulteriori informazioni, consulta Estendi il tuo spazio di lavoro con i plugin nella documentazione di Amazon Managed Grafana.	AWS DevOps, DevOps ingegnere

Visualizza le metriche personalizzate nella dashboard di Amazon Managed Grafana

Operazione	Description	Competenze richieste
Visualizza la dashboard di Amazon Managed Grafana.	Per visualizzare le metriche che sono state inserite nell'area di lavoro di Amazon Managed Grafana, procedi come segue: 1. Nella console Amazon Managed Grafana, scegli Dashboard nel riquadro di navigazione a sinistra. 2. Per visualizzare le metriche, scegli le dashboard degli eventi MWAA, quindi seleziona mwaa_metrics. La pagina delle metriche della dashboard mostra le seguenti informazioni: • Il DAG totale viene eseguito nell'ultima ora • Totale di esecuzioni DAG riuscite, non riuscite e in esecuzione nell'ultima ora • Durata media di tutte le esecuzioni DAG, riuscite e non riuscite	AWS DevOps
Personalizza la dashboard di Amazon Managed Grafana.	Per personalizzare i dashboard per ulteriori miglioramenti futuri, procedi come segue:	AWS DevOps

Operazione	Description	Competenze richieste
	1. Nella `mwaa_metrics` pagina della dashboard di Amazon Managed Grafana, scegli l'icona delle impostazioni della dashboard. 2. Per visualizzare la struttura dei dati che definisce la dashboard, scegli il modello JSON. Puoi personalizzare la dashboard apportando modifiche a questo modello JSON direttamente nella console. In alternativa, il codice sorgente di questa dashboard è disponibile nel `dashboard.json` file nella `stacks/infra/grafana` cartella del repository. GitHub	

Pulisci AWS le risorse

Operazione	Description	Competenze richieste
Metti in pausa le esecuzioni di Amazon MWAA DAG.	Per mettere in pausa le esecuzioni del DAG, procedi come segue: 1. Nella console Amazon MWAA, accedi agli ambienti Airflow e scegli Open Airflow UI.	AWS DevOps, ingegnere dei dati

Operazione	Description	Competenze richieste
	2. Per mettere in pausa il DAG, usa l'interruttore a levetta accanto a ciascun DAG. 3. Aggiorna la pagina dell'interfaccia utente Airflow, che dovrebbe elencarne tre nella sezione Pausa. DAGs	
Elimina gli oggetti nei bucket Amazon S3.	Per eliminare i bucket Amazon S3 mwaa-events-bucket-* e mwaa-metrics-bucket-*, segui le istruzioni per l'uso della console Amazon S3 in Eliminazione di un bucket nella documentazione di Amazon S3.	AWS DevOps

Operazione	Description	Competenze richieste
Distruggi le risorse create da Terraform.	Per distruggere le risorse create da Terraform e il file di stato locale Terraform associato, procedi come segue: 1. (Facoltativo) Prima di eliminare le risorse, puoi visualizzare in anteprima le modifiche che Terraform apporterà. Per generare un piano, esegui il seguente comando: `terraform plan -destroy` L'output del comando mostra che il `destroy` comando eliminerà tutte le AWS risorse create in precedenza. 2. `terraform destroy -auto-approve` Questo comando impiega circa 20 minuti per distruggere l'infrastruttura. **Note** Per distruggere tutte le risorse gestite da Terraform , esegui il seguente	AWS DevOps

Operazione	Description	Competenze richieste

	comando. : Il `-auto-approve` tag non attende la conferma dell'utente per iniziare a distruggere le risorse.
	3. Per eliminare il file di stato locale di Terraform, esegui i seguenti comandi:
	`rm .terraform.lock.hcl`
	`rm -rf .terraform`
	`rm terraform.tfstate*`	

risoluzione dei problemi

Problema	Soluzione
`null_resource.plugin_mgmt (local-exec): aws: error: argument operation: Invalid choice, valid choices are:`	Aggiorna il tuo AWS CLI alla versione più recente.
Errore durante il caricamento delle fonti di dati - `Fetch error: 404 Not Found Instantiating…`	L'errore è intermittente. Attendi qualche minuto, quindi aggiorna le sorgenti dati per visualizzare l'origine dati Timestream elencata.

Risorse correlate

AWS documentazione

- Amazon Managed Grafana per dashboard e visualizzazione
- Configura Amazon Managed Grafana per usare Okta
- Utilizzalo AWS IAM Identity Center con il tuo spazio di lavoro Amazon Managed Grafana
- Lavorare con DAGs Amazon MWAA

AWS video

- Configura IAM Identity Center con Amazon Managed Grafana per l'autenticazione, come mostrato nel video seguente.

https://www.youtube-nocookie.com/embed/XX2Xcz-Ps9U? controlli = 0

- Se IAM Identity Center non è disponibile, puoi anche integrare l'autenticazione Amazon Managed Grafana utilizzando un provider di identità (IdP) esterno come Okta, come mostrato nel video seguente.

https://www.youtube-nocookie.com/embed/A4 Og? JHxl2xp controlli = 0

Informazioni aggiuntive

Puoi creare una soluzione completa di monitoraggio e avviso per il tuo ambiente Amazon MWAA, che consente una gestione proattiva e una risposta rapida a potenziali problemi o anomalie. Amazon Managed Grafana include le seguenti funzionalità:

Avvisi: puoi configurare gli avvisi in Amazon Managed Grafana in base a soglie o condizioni predefinite. Imposta notifiche e-mail per avvisare le parti interessate quando determinate metriche superano o scendono al di sotto delle soglie specificate. Per ulteriori informazioni, consulta gli avvisi Grafana nella documentazione di Amazon Managed Grafana.

Integrazione: puoi integrare Amazon Managed Grafana con vari strumenti di terze parti come OpsGenie PagerDuty, o Slack per funzionalità di notifica avanzate. Ad esempio, puoi configurare webhook o integrarli per attivare incidenti e notifiche in queste piattaforme in base APIs agli avvisi generati in Amazon Managed Grafana. Inoltre, questo modello fornisce un GitHub archivio per creare risorse. AWS È possibile integrare ulteriormente questo codice con i flussi di lavoro di implementazione dell'infrastruttura.

Automatizza la distribuzione di stack set utilizzando AWS e AWS CodePipeline CodeBuild

Thiyagarajan Mani, Mihir Borkar e Raghu Gowda, Amazon Web Services

Riepilogo

Nella soluzione di integrazione continua e distribuzione continua (CI/CD) processes, you might want to deploy applications automatically into all your existing AWS accounts and into new accounts that you add to your organization in AWS Organizations. When you architect a CI/CDsoluzione per questo requisito), la funzionalità di amministratore di set di stack delegati di AWS CloudFormation è utile perché consente un livello di sicurezza limitando l'accesso all'account di gestione. Tuttavia, AWS CodePipeline utilizza il modello di autorizzazioni gestite dai servizi per distribuire applicazioni in più account e regioni. È necessario utilizzare l'account di gestione AWS Organizations per eseguire la distribuzione con set di stack, poiché AWS CodePipeline non supporta la funzionalità di amministratore delegato degli stack set.

Questo modello descrive come aggirare questa limitazione. Il modello utilizza AWS CodeBuild e uno script personalizzato per automatizzare la distribuzione di stack set con AWS. CodePipeline Automatizza queste attività di distribuzione delle applicazioni:

- Implementa un'applicazione come set di stack nelle unità organizzative esistenti () OUs
- Estendi la distribuzione di un'applicazione ad altre regioni OUs
- Rimuovi un'applicazione distribuita da tutte o da alcune OUs regioni

Prerequisiti e limitazioni

Prerequisiti

Prima di seguire i passaggi indicati in questo schema:

- Crea organizzazioni nel tuo account di gestione AWS Organizations. Per istruzioni, consulta la documentazione di AWS Organizations.
- Abilita l'accesso affidabile tra AWS Organizations e utilizza CloudFormation le autorizzazioni gestite dai servizi. Per istruzioni, consulta Enable trusted access with AWS Organizations nella CloudFormation documentazione.

Limitazioni

Il codice fornito con questo pattern presenta le seguenti limitazioni:

- È possibile distribuire un solo CloudFormation modello per un'applicazione; la distribuzione di più modelli non è attualmente supportata.
- La personalizzazione dell'implementazione corrente richiede DevOps esperienza.
- Questo modello non utilizza le chiavi AWS Key Management System (AWS KMS). Tuttavia, puoi abilitare questa funzionalità riconfigurando il CloudFormation modello incluso in questo pattern.

Architecture

Questa architettura per la pipeline di CI/CD implementazione gestisce quanto segue:

- Limita l'accesso diretto all'account di gestione delegando la responsabilità della distribuzione dello stack set a un CI/CD account dedicato in qualità di amministratore dello stack set per le distribuzioni delle applicazioni.
- Utilizza il modello di autorizzazione gestito dal servizio per distribuire automaticamente l'applicazione ogni volta che un nuovo account viene creato e mappato in un'unità organizzativa.
- Garantisce la coerenza delle versioni dell'applicazione su tutti gli account a livello di ambiente.
- Utilizza più fasi di approvazione a livello di repository e pipeline per fornire ulteriori livelli di sicurezza e governance per l'applicazione distribuita.
- Supera l'attuale limitazione CodePipeline utilizzando uno script di distribuzione personalizzato per distribuire o rimuovere automaticamente set di stack e CodeBuild istanze di stack. Per un'illustrazione del controllo del flusso e della gerarchia delle chiamate API implementate dallo script personalizzato, consulta la sezione Informazioni aggiuntive.
- Crea set di stack individuali per gli ambienti di sviluppo, test e produzione. Inoltre, puoi creare set di stack che combinano più OUs regioni in ogni fase. Ad esempio, è possibile combinare sandbox e sviluppo OUs in una fase di sviluppo e implementazione.
- Supporta la distribuzione o l'esclusione di applicazioni in un sottoinsieme di account o un elenco di. OUs

Automazione e scalabilità

Puoi utilizzare il codice fornito con questo pattern per creare un CodeCommit repository AWS e una pipeline di codice per la tua applicazione. Puoi quindi distribuirli come set di stack in più account a

livello di unità organizzativa. Il codice automatizza anche componenti come gli argomenti di Amazon Simple Notification Service (Amazon SNS) per notificare gli approvatori, i ruoli AWS Identity and Access Management (IAM) richiesti e la policy di controllo del servizio (SCP) da applicare all'account di gestione.

Tools (Strumenti)

Servizi AWS

- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account e regioni AWS.
- AWS CodeBuild è un servizio di build completamente gestito che ti aiuta a compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS CodeCommit è un servizio di controllo delle versioni che consente di archiviare e gestire archivi Git in modo privato, senza dover gestire il proprio sistema di controllo del codice sorgente.
- AWS CodeDeploy automatizza le distribuzioni su Amazon Elastic Compute Cloud (Amazon EC2) o istanze locali, funzioni AWS Lambda o servizi Amazon Elastic Container Service (Amazon ECS).
- AWS ti CodePipeline aiuta a modellare e configurare rapidamente le diverse fasi di un rilascio di software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- AWS Organizations è un servizio di gestione degli account che ti aiuta a consolidare più account AWS in un'organizzazione da creare e gestire centralmente.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.

Repository di codici

Il codice per questo pattern è disponibile nel repository GitHub automated-code-pipeline-stackset-deployment. Per la struttura delle cartelle e altri dettagli, consulta il file readme del repository.

Best practice

Questo modello limita l'accesso diretto all'account di gestione durante la distribuzione dell'applicazione a livello di unità organizzativa. L'aggiunta di più fasi di approvazione al processo di pipeline e repository contribuisce a fornire maggiore sicurezza e governance per le applicazioni e i componenti distribuiti utilizzando questo approccio.

Epiche

Configurazione degli account in AWS Organizations

Operazione	Description	Competenze richieste
Abilita tutte le funzionalità dell'account di gestione.	Abilita tutte le funzionalità dell'account di gestione per la tua organizzazione seguendo le istruzioni nella documentazione di AWS Organizations.	Amministratore AWS, amministratore della piattaforma
Crea un CI/CD account.	In AWS Organizations, nella tua organizzazione, crea un CI/CD account dedicato e assegna un team che possieda e controlli l'accesso all'account.	Amministratore AWS
Aggiungi un amministratore delegato.	Nell'account di gestione, registra l' CI/CD account creato nel passaggio precedente come amministratore delegato di stack set. Per istruzioni, consulta la CloudFormation documentazione AWS.	Amministratore AWS, amministratore della piattaforma

Crea un repository e CI/CD una pipeline di applicazioni

Operazione	Description	Competenze richieste
Clona il repository del codice.	1. Clona l'archivio di codice fornito con questo pattern sul tuo computer:	AWS DevOps

Operazione	Description	Competenze richieste
	`git clone https://github.com/aws-samples/automated-code-pipeline-stackset-deployment.git` 2. Esamina il file readme per comprendere la struttura delle cartelle e altri dettagli.	

Operazione	Description	Competenze richieste
Crea argomenti SNS.	Puoi utilizzare il `sns-template.yaml` modello fornito nel GitHub repository per creare argomenti SNS e configurare le sottoscrizioni per le richieste di approvazione. 1. Sulla console AWS, accedi all' CI/CD account. 2. Apri la CloudFormation console all'indirizzo https://console.aws.amazon.com/cloudformation. 3. Crea un nuovo stack con nuove risorse (opzione standard). 4. Per Specificare il modello, scegliete Carica un file modello, Scegli file, quindi selezionate il `sns-template.yaml` file dalla `templates` cartella del repository GitHub clonato. Scegli Next (Successivo). 5. Fornisci un nome significativo per lo stack di applicazioni. 6. Specificate un prefisso per le risorse. 7. Scegli Avanti, Avanti e Invia.	AWS DevOps

Operazione	Description	Competenze richieste
	8. Quando lo stack è stato creato correttamente, scegli la scheda Outputs e annota gli Amazon Resource Names (ARNs) degli argomenti SNS per le pull request, l'ambiente di test e l'ambiente di produzione. Utilizzerai queste informazioni nei passaggi successivi.	

Operazione	Description	Competenze richieste
Crea ruoli IAM per CI/CD i componenti.	Puoi utilizzare il `cicd-role-template.yaml` modello fornito nel GitHub repository per creare ruoli e policy IAM richiesti dai CI/CD component i. 1. Sulla console AWS, accedi all' CI/CD account. 2. Apri la CloudFormation console all'indirizzo https://console.aws.amazon.com/cloudformation. 3. Crea un nuovo stack con nuove risorse (opzione standard). 4. Per Specificare il modello, scegliete Carica un file modello, Scegli file, quindi selezionate il `cicd-role-template.yaml` file dalla `templates` cartella del repository GitHub clonato. Scegli Next (Successivo). 5. Fornisci un nome significativo per lo stack di applicazioni. 6. Immettete i valori per i seguenti parametri: • L'ARN per la politica dei limiti delle autorizzazioni. Puoi ottenere questo ARN dalla sezione	AWS DevOps

Operazione	Description	Competenze richieste
	Dettagli della politica sui limiti delle autorizzazioni sulla console IAM. • L'ARN per l'argomento di approvazione della produzione SNS che hai annotato in precedenza. • L'ARN per l'argomento di approvazione del test SNS che hai annotato in precedenza. • Un prefisso per le risorse create dal modello. 7. Scegli Avanti, Avanti e Invia. 8. Quando lo stack è stato creato correttamente, scegli la scheda Output e annota i ARNs ruoli IAM che sono stati creati. Utilizzerai queste informazioni nei passaggi successivi.	

Operazione	Description	Competenze richieste
Crea un CodeCommit repository e una pipeline di codice per la tua applicazione.	Puoi utilizzare il `cicd-pipeline-template.yaml` modello fornito nel repository per creare un GitHub CodeCommit repository e una pipeline di codice per la tua applicazione. 1. Sulla console AWS, accedi all' CI/CD account. 2. Apri la CloudFormation console all'indirizzo https://console.aws.amazon.com/cloudformation. 3. Crea un nuovo stack con nuove risorse (opzione standard). 4. Per Specificare il modello, scegliete Carica un file modello, Scegli file, quindi selezionate il `cicd-pipeline-template.yaml` file dalla `templates` cartella del repository GitHub clonato. Scegli Next (Successivo). 5. Fornisci un nome significativo per lo stack di applicazioni. 6. Immettete i valori per i seguenti parametri: • AppRepositoryName— Il nome del CodeCommi	AWS DevOps

Operazione	Description	Competenze richieste
	t repository che verrà creato per l'applicazione. • AppRepositoryDescription— Una breve descrizione del CodeCommit repository che verrà creato per l'applicazione. • ApplicationName— Il nome dell'applicazione. Questa stringa viene utilizzata come nome del CodeCommit repository e come prefisso della CI/CD pipeline. • CloudWatchEventRoleARN: l'ARN del ruolo dell' CloudWatch evento dell'attività precedente. • CodeBuildProjectRoleARN: l'ARN del ruolo del CodeBuild progetto nell'attività precedente. • CodePipelineRoleARN: l'ARN del CodePipeline ruolo dell'attività precedente. • DeploymentConfigBucket— Il nome del bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) in cui verranno archiviati i file	

Operazione	Description	Competenze richieste
	di configurazione della distribuzione e il file.zip dello script. • DeploymentConfigKey— Il percorso e il nome del file.zip (chiave Amazon S3). • PRApprovalSNSARN — L'ARN per l'argomento SNS per le notifiche di pull request. • ProdApprovalSNSARN : l'ARN per l'argomento SNS per le approvazioni di produzione. • TESTApprovalSNSARN : l'ARN per l'argomento SNS per le approvazioni dei test. • TemplateBucket— Il nome del bucket S3 nell' CI/CD account in cui verrà archiviato il modello di creazione della CI/CD pipeline. 7. Scegli Avanti, Avanti e Invia. 8. Quando lo stack viene completato correttam ente, crea un CodeCommi t repository con il nome specificato e una struttura di directory predefinita,	

Operazione	Description	Competenze richieste
	file di configurazione della distribuzione, script e una pipeline di codice per il repository.	

Implementa un set di stack

Operazione	Description	Competenze richieste

Clonare il repository dell'applicazione.	Il modello di CI/CD pipeline utilizzato in precedenza crea un repository di applicazioni di esempio e una pipeline di codice. Per clonare e verificare il repository:
	1. Accedi all'account. CI/CD
	2. Trova il repository e la CI/CD pipeline delle applicazioni che hai creato nell'epoca precedente.
	3. Copia l'URL del repository e usa il comando git clone per clonare il repository sul tuo computer locale.
	4. Verifica che la struttura della directory e i file corrispondano ai seguenti:
	`root`
	`|- deploy_configs`
	`|  |- deployment_config.json`
	`|- parameters`
		Sviluppatore di app, ingegnere dei dati

Operazione	Description	Competenze richieste

	<pre>| |- template-parameter-dev.json
	| |- template-parameter-test.json
	| |- template-parameter-prod.json
	|- templates
	| |- template.yml
	|- buildspec.yml</pre> dove la `deploy_configs` cartella contiene il file di configurazione della distribuzione e le `parameters` cartelle `templates` and includono i file predefiniti che sostituirete con i vostri file di CloudFormation modelli e parametri.
	Important
	Non personalizzate la struttura delle cartelle.
	5. Crea un feature branch.	

Operazione	Description	Competenze richieste
Aggiungi artefatti applicativi.	Aggiorna il repository dell'applicazione utilizzando un modello. CloudFormation **Note** Questa soluzione supporta la distribuzione di un solo CloudFormation modello. 1. Crea il tuo CloudFormation modello per distribuire le modifiche al codice dell'applicazione e assegnagli un `<application-name>.yaml` nome. 2. Sostituite il `template.yml` file nella `templates` cartella dell'archivio dell'applicazione con il CloudFormation modello creato nel passaggio 1. 3. Preparate i file dei parametri per ogni ambiente (sviluppo, test e produzione). 4. Assegnate un nome ai file dei parametri utilizzando il formato`<cloudformation-template-name>-parameter-<env`	Sviluppatore di app, ingegnere dei dati

Operazione	Description	Competenze richieste
	ironment-name>.jso n . 5. Sostituite i file dei parametri predefiniti nella parameter s cartella con i file del passaggio 4.	

Operazione	Description	Competenze richieste
Aggiorna il file di configurazione della distribuzione.	Aggiorna il `deploymen t_config.json` file: 1. Nel repository dell'applicazione, accedete alla `deploy_configs` cartella. 2. Apri il file`deploymen t_config.json` : `{ "deployme nt_action": "<deploy/delete>", "stack_set_name": "<stack set name>", "stack_se t_desciption": "<stack set descripti on>", "deployme nt_targets": { "dev": { "org_units": ["list of OUs"], "regions": ["list of regions"], "filter_accounts": ["list of accounts" ], "filter_type":`	Sviluppatore di app, ingegnere dei dati

Operazione	Description	Competenze richieste

	"<DIFFERENCE/INTERSECTION/UNION>"
	},
	"test": {
	"org_units": ["list of OUs"],
	"regions": ["list of regions"],
	"filter_accounts": ["list of accounts"],
	"filter_type": "<DIFFERENCE/INTERSECTION/UNION>"
	},
	"prod": {
	"org_units": ["list of OUs"],
	"regions": ["list of regions"],
	"filter_accounts": ["list of accounts"],	

Operazione	Description	Competenze richieste
	"filter_type": "<DIFFERENCE/INTERSECTION/UNION>" } }, "cft_capabilities": ["CAPABILITY_IAM", "CAPABILITY_NAMED_IAM"], "auto_deployement": "<True/False>", "retain_stacks_on_account_removal": "<True/False>", "region_deployment_concurrency": "<SEQUENTIAL/PARALLEL>" }	
	3. Aggiorna i valori per l'azione di distribuzione, il nome del set di stack, la descrizione dello stack set e gli obiettivi di distribuzione. Ad esempio, è possibile impostare l'eliminazione dell'intero set di stack e delle relative istanze di stack associate. `deploymen t_action delete` `deploy`Utilizzatelo per	

Operazione	Description	Competenze richieste
	creare un nuovo set di stack, per aggiornare un set di stack esistente o per aggiungere o rimuovere istanze di stack per altre regioni. OUs Per altri esempi, consulta la sezione Informazioni aggiuntive. Questo modello crea set di stack individuali per ogni ambiente aggiungendo il nome dell'ambiente al nome dello stack set fornito nel file di configurazione della distribuzione.	

Operazione	Description	Competenze richieste
Applica le modifiche e distribuisci lo stack set.	Applica le modifiche specificate nel modello dell'applicazione, quindi unisci e distribuisci lo stack set in più ambienti fase per fase: 1. Salvate tutti i file e salvate le modifiche nel ramo delle funzionalità del repository locale dell'applicazione. 2. Invia il ramo delle funzionalità al repository remoto. 3. Crea una pull request per unire le modifiche al ramo principale. Quando la pull request è stata approvata e le modifiche sono state unite al ramo principale, verrà avviata la CI/CD pipeline. 4. Una volta completata con successo la fase di sviluppo, controlla la CloudFormation console, scheda Service-Managed. StackSets Vedrai un nuovo set di stack con il suffisso. `dev` 5. Controlla i CodeBuild log della fase di sviluppo per eventuali problemi.	Sviluppatore di app, ingegnere dei dati

Operazione	Description	Competenze richieste
	6. Implementa lo stack impostato negli ambienti di test e produzione chiedendo agli approvatori di approvare le implement azioni per quelle fasi e ripetendo i passaggi 5 e 6. I set di stack per gli ambienti di test e produzione hanno i suffissi e. `test` `prod`	

risoluzione dei problemi

Problema	Soluzione
L'implementazione fallisce con l'eccezione: Cambia il nome del file dei parametri del modello come -parameter- .json con, i nomi predefiniti non sono consentiti <application name><evn>	I file dei parametri del CloudFormation modello devono seguire la convenzione di denominaz ione specificata. Aggiorna i nomi dei file dei parametri e riprova.
La distribuzione non riesce con l'eccezione: Modificare il nome del CloudFormation modello in .yml, i valori predefiniti template.yml o template.yaml non sono consentiti <application name>	CloudFormation Il nome del modello deve seguire la convenzione di denominazione specificata. Aggiorna il nome del file e riprova.
La distribuzione non riesce con l'eccezione: Nessun CloudFormation modello valido e il relativo file di parametri trovati per l'ambiente {environment name}	Controlla le convenzioni di denominazione dei file per il CloudFormation modello e il relativo file dei parametri per l'ambiente specificato.

Problema	Soluzione
La distribuzione fallisce con l'eccezione: Azione di distribuzione non valida fornita nel file di configurazione della distribuzione. Le opzioni valide sono 'deploy' e 'delete'.	Hai specificato un valore non valido per il `deployment_action` parametro nel file di configurazione della distribuzione. Il parametro ha due valori validi: `deploy` e`delete`. Utilizzato `deploy` per creare e aggiornare i set di stack e le relative istanze di stack associate . `delete`Utilizzatelo solo quando desiderate rimuovere l'intero set di stack e le istanze di stack associate.

Risorse correlate

- GitHub automated-code-pipeline-stackset-repository di distribuzione
- Abilitazione di tutte le funzionalità dell'organizzazione (documentazione AWS Organizations)
- Registrare un amministratore delegato (CloudFormation documentazione AWS)
- Obiettivi a livello di account per set di stack gestiti dai servizi (documentazione AWS) CloudFormation

Informazioni aggiuntive

diagramma di flusso

Il seguente diagramma di flusso illustra il controllo del flusso e la gerarchia delle chiamate API implementate dallo script personalizzato per automatizzare la distribuzione degli stack set.

Esempi di file di configurazione della distribuzione

Creazione di un nuovo set di stack

Il seguente file di configurazione della distribuzione crea un nuovo stack set chiamato `sample-stack-set` nella regione AWS `us-east-1` in tre OUs.

```
{
     "deployment_action": "deploy",
     "stack_set_name": "sample-stack-set",
```

```
     "stack_set_desciption": "this is a sample stack set",
    "deployment_targets": {
                            "dev": {
                                        "org_units": ["dev-org-unit-1"],
                                        "regions": ["us-east-1"],
                                        "filter_accounts": [],
                                        "filter_type": ""
                                    },
                            "test": {
                                        "org_units": ["test-org-unit-1"],
                                        "regions": ["us-east-1"],
                                        "filter_accounts": [],
                                        "filter_type": ""
                                    },
                            "prod": {
                                        "org_units": ["prod-org-unit-1"],
                                        "regions": ["us-east-1"],
                                        "filter_accounts": [],
                                        "filter_type": ""
                                    }
                           },
     "cft_capabilities": ["CAPABILITY_IAM", "CAPABILITY_NAMED_IAM"],
     "auto_deployement": "True",
     "retain_stacks_on_account_removal": "True",
     "region_deployment_concurrency": "PARALLEL"
 }
```

Distribuzione di uno stack set esistente su un'altra unità organizzativa

Se si distribuisce la configurazione mostrata nell'esempio precedente e si desidera distribuire lo stack set su un'unità organizzativa aggiuntiva chiamata `dev-org-unit-2` nell'ambiente di sviluppo, il file di configurazione della distribuzione potrebbe avere il seguente aspetto.

```
{
     "deployment_action": "deploy",
     "stack_set_name": "sample-stack-set",
     "stack_set_desciption": "this is a sample stack set",
    "deployment_targets": {
                            "dev": {
                                        "org_units": ["dev-org-unit-1", "dev-org-
unit-2"],
                                        "regions": ["us-east-1"],
                                        "filter_accounts": [],
```

```
                                    "filter_type": ""
                                },
                            "test": {
                                    "org_units": ["test-org-unit-1"],
                                    "regions": ["us-east-1"],
                                    "filter_accounts": [],
                                    "filter_type": ""
                                },
                            "prod": {
                                    "org_units": ["prod-org-unit-1"],
                                    "regions": ["us-east-1"],
                                    "filter_accounts": [],
                                    "filter_type": ""
                                }
                        },
    "cft_capabilities": ["CAPABILITY_IAM", "CAPABILITY_NAMED_IAM"],
    "auto_deployement": "True",
    "retain_stacks_on_account_removal": "True",
    "region_deployment_concurrency": "PARALLEL"
}
```

Implementazione di uno stack esistente impostato in un'altra regione AWS

Se distribuisci la configurazione mostrata nell'esempio precedente e desideri distribuire lo stack impostato in una regione AWS aggiuntiva (`us-east-2`) nell'ambiente di sviluppo per due OUs (`dev-org-unit-1edev-org-unit-2`), il file di configurazione della distribuzione potrebbe essere simile al seguente.

> **Note**
>
> Le risorse nel CloudFormation modello devono essere valide e specifiche della regione.

```
{
    "deployment_action": "deploy",
    "stack_set_name": "sample-stack-set",
    "stack_set_desciption": "this is a sample stack set",
   "deployment_targets": {
                            "dev": {
                                    "org_units": ["dev-org-unit-1", "dev-org-
unit-2"],
                                    "regions": ["us-east-1", "us-east-2"],
```

```
                                    "filter_accounts": [],
                                    "filter_type": ""
                                },
                            "test": {
                                    "org_units": ["test-org-unit-1"],
                                    "regions": ["us-east-1"],
                                    "filter_accounts": [],
                                    "filter_type": ""
                                },
                            "prod": {
                                    "org_units": ["prod-org-unit-1"],
                                    "regions": ["us-east-1"],
                                    "filter_accounts": [],
                                    "filter_type": ""
                                }
                        },
    "cft_capabilities": ["CAPABILITY_IAM", "CAPABILITY_NAMED_IAM"],
    "auto_deployement": "True",
    "retain_stacks_on_account_removal": "True",
    "region_deployment_concurrency": "PARALLEL"
}
```

Rimozione di un'istanza stack da un'unità organizzativa o da una regione AWS

Supponiamo che la configurazione di distribuzione mostrata nell'esempio precedente sia stata implementata. Il seguente file di configurazione rimuove le istanze dello stack da entrambe le regioni dell'unità organizzativa. `dev-org-unit-2`

```
{
    "deployment_action": "deploy",
    "stack_set_name": "sample-stack-set",
    "stack_set_desciption": "this is a sample stack set",
   "deployment_targets": {
                            "dev": {
                                    "org_units": ["dev-org-unit-1"],
                                    "regions": ["us-east-1", "us-east-2"],
                                    "filter_accounts": [],
                                    "filter_type": ""
                                },
                            "test": {
                                    "org_units": ["test-org-unit-1"],
                                    "regions": ["us-east-1"],
                                    "filter_accounts": [],
```

```
                                        "filter_type": ""
                                    },
                                "prod": {
                                        "org_units": ["prod-org-unit-1"],
                                        "regions": ["us-east-1"],
                                        "filter_accounts": [],
                                        "filter_type": ""
                                    }
                            },
    "cft_capabilities": ["CAPABILITY_IAM", "CAPABILITY_NAMED_IAM"],
    "auto_deployement": "True",
    "retain_stacks_on_account_removal": "True",
    "region_deployment_concurrency": "PARALLEL"
 }
```

Il seguente file di configurazione rimuove l'istanza dello stack dalla regione AWS `us-east-1` per entrambe OUs nell'ambiente di sviluppo.

```
{
     "deployment_action": "deploy",
     "stack_set_name": "sample-stack-set",
     "stack_set_desciption": "this is a sample stack set",
    "deployment_targets": {
                                "dev": {
                                        "org_units": ["dev-org-unit-1", "dev-org-
unit-2"],
                                        "regions": ["us-east-2"],
                                        "filter_accounts": [],
                                        "filter_type": ""
                                    },
                                "test": {
                                        "org_units": ["test-org-unit-1"],
                                        "regions": ["us-east-1"],
                                        "filter_accounts": [],
                                        "filter_type": ""
                                    },
                                "prod": {
                                        "org_units": ["prod-org-unit-1"],
                                        "regions": ["us-east-1"],
                                        "filter_accounts": [],
                                        "filter_type": ""
                                    }
```

```
                            },
     "cft_capabilities": ["CAPABILITY_IAM", "CAPABILITY_NAMED_IAM"],
     "auto_deployement": "True",
     "retain_stacks_on_account_removal": "True",
     "region_deployment_concurrency": "PARALLEL"
 }
```

Eliminazione dell'intero set di stack

Il seguente file di configurazione di distribuzione elimina l'intero set di stack e tutte le istanze di stack associate.

```
{
     "deployment_action": "delete",
     "stack_set_name": "sample-stack-set",
     "stack_set_desciption": "this is a sample stack set",
    "deployment_targets": {
                             "dev": {
                                        "org_units": ["dev-org-unit-1", "dev-org-
unit-2"],
                                        "regions": ["us-east-2"],
                                        "filter_accounts": [],
                                        "filter_type": ""
                                    },
                             "test": {
                                        "org_units": ["test-org-unit-1"],
                                        "regions": ["us-east-1"],
                                        "filter_accounts": [],
                                        "filter_type": ""
                                    },
                             "prod": {
                                        "org_units": ["prod-org-unit-1"],
                                        "regions": ["us-east-1"],
                                        "filter_accounts": [],
                                        "filter_type": ""
                                    }
                            },
     "cft_capabilities": ["CAPABILITY_IAM", "CAPABILITY_NAMED_IAM"],
     "auto_deployement": "True",
     "retain_stacks_on_account_removal": "True",
     "region_deployment_concurrency": "PARALLEL"
 }
```

Esclusione di un account dalla distribuzione

Il seguente file di configurazione di distribuzione esclude l'account`111122223333`, che fa parte dell'unità organizzativa`dev-org-unit-1`, dalla distribuzione.

```
{
    "deployment_action": "deploy",
    "stack_set_name": "sample-stack-set",
    "stack_set_desciption": "this is a sample stack set",
   "deployment_targets": {
                           "dev": {
                                       "org_units": ["dev-org-unit-1"],
                                       "regions": ["us-east-1"],
                                       "filter_accounts": ["111122223333"],
                                       "filter_type": "DIFFERENCE"
                                   },
                           "test": {
                                       "org_units": ["test-org-unit-1"],
                                       "regions": ["us-east-1"],
                                       "filter_accounts": [],
                                       "filter_type": ""
                                   },
                           "prod": {
                                       "org_units": ["prod-org-unit-1"],
                                       "regions": ["us-east-1"],
                                       "filter_accounts": [],
                                       "filter_type": ""
                                   }
                        },
    "cft_capabilities": ["CAPABILITY_IAM", "CAPABILITY_NAMED_IAM"],
    "auto_deployement": "True",
    "retain_stacks_on_account_removal": "True",
    "region_deployment_concurrency": "PARALLEL"
}
```

Distribuzione dell'applicazione su un sottoinsieme di account in un'unità organizzativa

Il seguente file di configurazione di distribuzione distribuisce l'applicazione solo su tre account (`111122223333444455556666`, `e777788889999`) nell'unità organizzativa. `dev-org-unit-1`

```
{
    "deployment_action": "deploy",
    "stack_set_name": "sample-stack-set",
```

```
    "stack_set_desciption": "this is a sample stack set",
   "deployment_targets": {
                           "dev": {
                                       "org_units": ["dev-org-unit-1"],
                                       "regions": ["us-east-1"],
                                       "filter_accounts": ["111122223333",
"444455556666", "777788889999"],
                                       "filter_type": "INTERSECTION"
                                   },
                           "test": {
                                       "org_units": ["test-org-unit-1"],
                                       "regions": ["us-east-1"],
                                       "filter_accounts": [],
                                       "filter_type": ""
                                   },
                           "prod": {
                                       "org_units": ["prod-org-unit-1"],
                                       "regions": ["us-east-1"],
                                       "filter_accounts": [],
                                       "filter_type": ""
                                   }
                        },
    "cft_capabilities": ["CAPABILITY_IAM", "CAPABILITY_NAMED_IAM"],
    "auto_deployement": "True",
    "retain_stacks_on_account_removal": "True",
    "region_deployment_concurrency": "PARALLEL"
}
```

Associa automaticamente una policy gestita da AWS per Systems Manager ai profili di EC2 istanza utilizzando Cloud Custodian e AWS CDK

Ali Asfour e Aaron Lennon, Amazon Web Services

Riepilogo

Puoi integrare le istanze Amazon Elastic Compute Cloud (Amazon EC2) con AWS Systems Manager per automatizzare le attività operative e fornire maggiore visibilità e controllo. Per integrarsi con Systems Manager, EC2 le istanze devono avere una policy AWS Systems Manager Agent (SSM Agent) installata e una policy `AmazonSSMManagedInstanceCore` AWS Identity and Access Management (IAM) allegata ai profili delle istanze.

Tuttavia, se vuoi assicurarti che tutti i profili di EC2 istanza abbiano la `AmazonSSMManagedInstanceCore` policy allegata, puoi affrontare problemi nell'aggiornamento di nuove EC2 istanze che non dispongono di profili di istanza o EC2 istanze che hanno un profilo di istanza ma non dispongono della policy. `AmazonSSMManagedInstanceCore` Inoltre, può essere difficile aggiungere questa policy su più account Amazon Web Services (AWS) e regioni AWS.

Questo modello aiuta a risolvere queste sfide implementando tre policy Cloud Custodian nei tuoi account AWS:

- La prima policy di Cloud Custodian verifica EC2 le istanze esistenti che hanno un profilo di istanza ma non dispongono della policy. `AmazonSSMManagedInstanceCore` La `AmazonSSMManagedInstanceCore` policy viene quindi allegata.
- La seconda policy di Cloud Custodian verifica EC2 le istanze esistenti senza un profilo di istanza e aggiunge un profilo di istanza predefinito a cui è associata la `AmazonSSMManagedInstanceCore` policy.
- La terza policy Cloud Custodian crea funzioni AWS Lambda nei tuoi account per monitorare la creazione EC2 di istanze e profili di istanze. Ciò garantisce che la `AmazonSSMManagedInstanceCore` policy venga allegata automaticamente quando viene creata un' EC2 istanza.

Questo modello utilizza DevOps gli strumenti AWS per ottenere una distribuzione continua e su larga scala delle policy di Cloud Custodian in un ambiente multi-account, senza fornire un ambiente di calcolo separato.

Prerequisiti e limitazioni

Prerequisiti

- Due o più account AWS attivi. Un account è l'account di sicurezza e gli altri sono account membri.
- Autorizzazioni per il provisioning di risorse AWS nell'account di sicurezza. Questo modello utilizza le autorizzazioni di amministratore, ma è necessario concedere le autorizzazioni in base ai requisiti e alle politiche dell'organizzazione.
- Capacità di assumere un ruolo IAM dall'account di sicurezza agli account dei membri e creare i ruoli IAM richiesti. Per ulteriori informazioni su questo argomento, consulta Delegare l'accesso tra account AWS utilizzando i ruoli IAM nella documentazione IAM.
- **Important**

  AWS Command Line Interface (AWS CLI), installata e configurata. A scopo di test, puoi configurare AWS CLI utilizzando il `aws configure` comando o impostando le variabili di ambiente. : Questa opzione non è consigliata per gli ambienti di produzione e consigliamo di concedere a questo account solo l'accesso con il minimo privilegio. Per ulteriori informazioni su questo argomento, consulta Concedere il privilegio minimo nella documentazione IAM.
- Il `devops-cdk-cloudcustodian.zip` file (allegato), scaricato sul computer locale.
- Familiarità con Python.
- Gli strumenti richiesti (Node.js, AWS Cloud Development Kit (AWS CDK) e Git), installati e configurati. Puoi utilizzare il `install-prerequisites.sh` file contenuto nel `devops-cdk-cloudcustodian.zip` file per installare questi strumenti. Assicurati di eseguire questo file con i privilegi di root.

Limitazioni

- Sebbene questo modello possa essere utilizzato in un ambiente di produzione, assicurati che tutti i ruoli e le policy IAM soddisfino i requisiti e le policy della tua organizzazione.

Versioni del pacchetto

- Cloud Custodian versione 0.9 o successiva
- TypeScript versione 3.9.7 o successiva

- Node.js versione 14.15.4 o successiva
- npmversione 7.6.1 o successiva
- AWS CDK versione 1.96.0 o successiva

Architecture

Il diagramma mostra il flusso di lavoro seguente:

1. Le policy di Cloud Custodian vengono trasferite in un CodeCommit repository AWS nell'account di sicurezza. Una regola di Amazon CloudWatch Events avvia automaticamente la CodePipeline pipeline AWS.
2. La pipeline recupera il codice più recente CodeCommit e lo invia alla parte di integrazione continua della pipeline di integrazione continua e distribuzione continua (CI/CD) gestita da AWS. CodeBuild
3. CodeBuild esegue le DevSecOps azioni complete, inclusa la convalida della sintassi delle policy sulle policy di Cloud Custodian, ed esegue queste policy in modalità per verificare quali risorse vengono identificate. `--dryrun`
4. Se non ci sono errori, l'attività successiva avvisa un amministratore di rivedere le modifiche e approvare la distribuzione negli account dei membri.

Stack tecnologico

- AWS CDK
- CodeBuild
- CodeCommit
- CodePipeline
- IAM
- Cloud Custodian

Automazione e scalabilità

Il modulo AWS CDK pipelines fornisce una CI/CD pipeline che utilizza CodePipeline per orchestrare la creazione e il test del codice sorgente CodeBuild, oltre alla distribuzione delle risorse AWS con gli stack AWS. CloudFormation Puoi utilizzare questo modello per tutti gli account dei membri e le

regioni della tua organizzazione. Puoi anche estendere lo `Roles creation` stack per distribuire altri ruoli IAM nei tuoi account membro.

Tools (Strumenti)

- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software per definire l'infrastruttura cloud nel codice e fornirla tramite AWS. CloudFormation
- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire con i servizi AWS utilizzando i comandi nella shell della riga di comando.
- AWS CodeBuild è un servizio di build completamente gestito nel cloud.
- AWS CodeCommit è un servizio di controllo delle versioni che puoi utilizzare per archiviare e gestire risorse in modo privato.
- AWS CodePipeline è un servizio di distribuzione continua che puoi utilizzare per modellare, visualizzare e automatizzare i passaggi necessari per rilasciare il tuo software.
- AWS Identity and Access Management è un servizio Web che ti aiuta a controllare in modo sicuro l'accesso alle risorse AWS.
- Cloud Custodian è uno strumento che riunisce le dozzine di strumenti e script utilizzati dalla maggior parte delle organizzazioni per gestire i propri account cloud pubblici in un unico strumento open source.
- Node.js è un JavaScript runtime basato sul motore V8 di Google Chrome. JavaScript

Codice

Per un elenco dettagliato dei moduli, delle funzioni dell'account, dei file e dei comandi di distribuzione utilizzati in questo modello, consultate il `README` file nel `devops-cdk-cloudcustodian.zip` file (allegato).

Epiche

Configura la pipeline con AWS CDK

Operazione	Description	Competenze richieste
Configura il CodeCommit repository.	1. Decomprimi il `devops-cdk-cloudcustodian.zip` file (allegato) nella	Developer

Operazione	Description	Competenze richieste
	directory di lavoro sul tuo computer locale. 2. Accedi alla Console di gestione AWS per il tuo account di sicurezza, apri la CodeCommit console e crea un nuovo `devops-cdk-cloudcustodian` repository. 3. Passa alla directory del progetto e configura il CodeCommit repository come origine, conferma le modifiche e poi inviale al ramo di origine eseguendo i seguenti comandi: • `cd devops-cdk-cloudcustodian` • `git init --initial-branch=main` • `git add . git commit -m 'initial commit'` • `git remote add origin https://git-codecommit.us-east-1.amazonaws.com/v1/devops-cdk-cloudcustodian` • `git push origin main` Per ulteriori informazioni su questo argomento, consulta	

Operazione	Description	Competenze richieste
	Creazione di un CodeCommit repository nella CodeCommit documentazione AWS.	
Installa gli strumenti richiesti.	Usa il `install-prerequisites.sh` file per installare tutti gli strumenti necessari su Amazon Linux. Questo non include la CLI di AWS perché è preinstallata. Per ulteriori informazioni su questo argomento, consulta la sezione Prerequisiti di Getting started with the AWS CDK nella documentazione di AWS CDK.	Developer

Operazione	Description	Competenze richieste
Installa i pacchetti AWS CDK richiesti.	1. Configura il tuo ambiente virtuale eseguendo il seguente comando nella CLI di AWS: `$ python3 -m venv .env` 2. Attiva il tuo ambiente virtuale eseguendo il seguente comando: `$ source .env/bin/activate` 3. Dopo l'attivazione dell'ambiente virtuale, installa le dipendenze richieste eseguendo il comando seguente: `$ pip install -r requirements.txt` 4. Per aggiungere dipendenze aggiuntive (ad esempio, altre librerie AWS CDK), aggiungile al `requirements.txt` file, quindi esegui il comando seguente: `pip install -r requirements.txt` I seguenti pacchetti sono richiesti da AWS CDK e sono inclusi nel `requirements.txt` file: • `aws-cdk.aws-cloudwatch`	Developer

Operazione	Description	Competenze richieste
	• `aws-cdk.aws-codebuild` • `aws-cdk.aws-codecommit` • `aws-cdk.aws-codedeploy` • `aws-cdk.aws-codepipeline` • `aws-cdk.aws-codepipeline-actions` • `aws-cdk.aws-events` • `aws-cdk.aws-events-targets` • `aws-cdk.aws-iam` • `aws-cdk.aws-logs` • `aws-cdk.aws-s3` • `aws-cdk.aws-sns` • `aws-cdk.aws-sns-subscriptions` • `aws-cdk.aws-sqs` • `aws-cdk.core`	

Configura il tuo ambiente

Operazione	Description	Competenze richieste
Aggiorna le variabili richieste.	Apri il `vars.py` file nella cartella principale del tuo CodeCommit repository e aggiorna le seguenti variabili:	Developer

Operazione	Description	Competenze richieste
	• Esegui l'aggiornamento `var_deploy_region = ‘us-east-1’` con la regione AWS in cui desideri che venga distribuita la pipeline. • Aggiorna `var_codecommit_repo_name = "cdk-cloudcustodian"` con il nome del tuo CodeCommit repository. • Aggiorna `var_codecommit_branch_name = "main"` con il nome del CodeCommit ramo. • Aggiorna `var_adminEmail=notifyadmin@email.com’` con l'indirizzo e-mail dell'amministratore che approva le modifiche. • Aggiorna `var_slackWebHookUrl = https://hooks.slack.com/services/T000000000/B00000000/XXXXXXXXXXXXXXXXXXXXXXXX` 'con il webhook Slack utilizzato per inviare notifiche a Cloud Custodian quando vengono apportate modifiche. • Aggiorna `var_orgId = ‘o-yyyyyyyyyy’` con l'ID della tua organizzazione.	

Operazione	Description	Competenze richieste
	• Aggiorna `security_account = '123456789011'` con l'ID dell'account AWS per l'account in cui viene distribuita la pipeline. • Esegui l'aggiornamento `member_accounts = ['111111111111','111111111112','111111111113']` con gli account membro in cui desideri avviare lo stack CDK AWS e distribuire i ruoli IAM richiesti. • Imposta su `cdk_bootstrap_member_accounts = True` `True` se desideri che la pipeline avvii automaticamente l'AWS CDK sui tuoi account membro. Se impostato su `True` questo valore, richiede anche il nome di un ruolo IAM esistente negli account dei membri, che può essere assunto dall'account di sicurezza. Questo ruolo IAM deve inoltre disporre delle autorizzazioni necessarie per avviare il CDK AWS.	

Operazione	Description	Competenze richieste
	• **Note** Aggiorna `cdk_bootstrap_role = 'AWSControlTowerExecution'` con il ruolo IAM esistente negli account dei membri che può essere assunto dall'account di sicurezza. Questo ruolo deve inoltre essere autorizzato a eseguire il bootstrap del CDK AWS. : Questo vale solo se `cdk_bootstrap_member_accounts` è impostato su. `True`	

Operazione	Description	Competenze richieste

Aggiorna il file account.yml con le informazioni sull'acco unt del membro.	Per eseguire lo strumento c7n-org Cloud Custodian su più account, è necessario inserire il file di `accounts.yml` configurazione nella radice del repository. Di seguito è riportato un esempio di file di configurazione di Cloud Custodian per AWS:
	`accounts:`
	`- account_id: '12312312 3123'`
	`name: account-1`
	`regions:`
	`- us-east-1`
	`- us-west-2`
	`role: arn:aws:i am::123123123123:r ole/CloudCustodian`
	`vars:`
	`charge_code: xyz`
	`tags:`
	`- type:prod`
	`- division:some division`
	`- partition:us`
	`- scope:pci`
		Developer

Esegui il bootstrap degli account AWS

Operazione	Description	Competenze richieste
Potenzia l'account di sicurezza .	Avvia il programma `deploy_account` con l'`cloudcustodian_sta`	Developer

Operazione	Description	Competenze richieste
	`ck` applicazione eseguendo il seguente comando: `cdk bootstrap -a 'python3 cloudcustodian/cloudcustodian_stack.py`	
Opzione 1 - Avvia automaticamente gli account dei membri.	Se la `cdk_bootstrap_member_accounts` variabile è impostata su `True` nel `vars.py` file, gli account specificati nella `member_accounts` variabile vengono automaticamente avviati dalla pipeline. Se necessario, puoi eseguire l'aggiornamento `*cdk_bootstrap_role*` con un ruolo IAM che puoi assumere dall'account di sicurezza e che dispone delle autorizzazioni necessarie per avviare il CDK AWS. I nuovi account aggiunti alla `member_accounts` variabile vengono avviati automaticamente dalla pipeline in modo da poter distribuire i ruoli richiesti.	Developer

Operazione	Description	Competenze richieste
Opzione 2: avvia manualmente gli account dei membri.	Sebbene non sia consigliabile utilizzare questo approccio, puoi impostare il valore di `cdk_bootstrap_member_accounts` to `False` ed eseguire questo passaggio manualmente eseguendo il comando seguente: `$ cdk bootstrap -a 'python3 cloudcustodian/member_account_roles_stack.py' \ --trust {security_account_id} \ --context assume-role-credentials:writeIamRoleName={role_name} \ --context assume-role-credentials:readIamRoleName={role_name} \ --mode=ForWriting \ --context bootstrap=true \ --cloudformation-execution-policies arn:aws:iam::aws:policy/AdministratorAccess`	Developer

Operazione	Description	Competenze richieste
	Important Assicurati di aggiornare `{role_name}` i valori `{security_account_id}` and con il nome di un ruolo IAM che puoi assumere dall'account di sicurezza e che disponga delle autorizzazioni necessarie per avviare il CDK AWS. Puoi anche utilizzare altri approcci per avviare gli account dei membri, ad esempio con AWS CloudFormation. Per ulteriori informazioni su questo argomento, consulta Bootstrapping nella documentazione di AWS CDK.	

Implementa gli stack CDK AWS

Operazione	Description	Competenze richieste
Crea i ruoli IAM negli account dei membri.	Esegui il comando seguente per distribuire lo `member_account_roles_stack` stack e creare i ruoli IAM negli account dei membri:	Developer

Operazione	Description	Competenze richieste
	`cdk deploy --all -a 'python3 cloudcustodian/member_account_roles_stack.py' --require-approval never`	
Implementa lo stack di pipeline Cloud Custodian.	Esegui il comando seguente per creare la `cloudcustodian_stack.py` pipeline Cloud Custodian da distribuire nell'account di sicurezza: `cdk deploy -a 'python3 cloudcustodian/cloudcustodian_stack.py'`	Developer

Risorse correlate

- Guida introduttiva alla CDK AWS

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Crea automaticamente CI/CD pipeline e cluster Amazon ECS per microservizi utilizzando AWS CDK

Varsha Raju, Amazon Web Services

Riepilogo

Questo modello descrive come creare automaticamente l'integrazione continua e la distribuzione continua (CI/CD) pipelines and underlying infrastructure for building and deploying microservices on Amazon Elastic Container Service (Amazon ECS). You can use this approach if you want to set up proof-of-concept CI/CD pipelines to show your organization the benefits of CI/CD, microservices, and DevOps. You can also use this approach to create initial CI/CDpipeline) che è quindi possibile personalizzare o modificare in base ai requisiti dell'organizzazione.

L'approccio del modello crea un ambiente di produzione e un ambiente non di produzione che dispongono ciascuno di un cloud privato virtuale (VPC) e di un cluster Amazon ECS configurati per l'esecuzione in due zone di disponibilità. Questi ambienti sono condivisi da tutti i tuoi microservizi e quindi crei una CI/CD pipeline per ogni microservizio. Queste CI/CD pipeline estraggono le modifiche da un repository di origine in AWS CodeCommit, creano automaticamente le modifiche e quindi le distribuiscono nei tuoi ambienti di produzione e non di produzione. Quando una pipeline completa con successo tutte le sue fasi, è possibile utilizzarla URLs per accedere al microservizio negli ambienti di produzione e non di produzione.

Prerequisiti e limitazioni

Prerequisiti

- Un account Amazon Web Services (AWS) attivo.
- Un bucket Amazon Simple Storage Service (Amazon S3) esistente che contiene `starter-code.zip` il file (allegato).
- AWS Cloud Development Kit (AWS CDK), installato e configurato nel tuo account. Per ulteriori informazioni su questo argomento, consulta Getting started with the AWS CDK nella documentazione di AWS CDK.
- Python 3 `epip`, installato e configurato. Per ulteriori informazioni su questo argomento, consulta la documentazione di Python.
- Familiarità con AWS CDK, AWS CodeBuild, CodePipeline AWS, CodeCommit Amazon Elastic Container Registry (Amazon ECR), Amazon ECS e AWS Fargate.
- Familiarità con Docker.

- Comprensione di e. CI/CD DevOps

Limitazioni

- Si applicano i limiti generali dell'account AWS. Per ulteriori informazioni a riguardo, consulta le quote dei servizi AWS nella documentazione di AWS General Reference.

Versioni del prodotto

- Il codice è stato testato utilizzando Node.js versione 16.13.0 e AWS CDK versione 1.132.0.

Architecture

Il diagramma mostra il flusso di lavoro seguente:

1. Uno sviluppatore di applicazioni inserisce il codice in un repository. CodeCommit
2. Viene avviata una pipeline.
3. CodeBuild crea e invia l'immagine Docker a un repository Amazon ECR
4. CodePipeline distribuisce una nuova immagine su un servizio Fargate esistente in un cluster Amazon ECS non di produzione.
5. Amazon ECS inserisce l'immagine dal repository Amazon ECR in un servizio Fargate non di produzione.
6. Il test viene eseguito utilizzando un URL non di produzione.
7. Il release manager approva la distribuzione di produzione.
8. CodePipeline distribuisce la nuova immagine su un servizio Fargate esistente in un cluster Amazon ECS di produzione
9. Amazon ECS inserisce l'immagine dal repository Amazon ECR nel servizio Fargate di produzione.
10. Gli utenti di produzione accedono alla funzionalità utilizzando un URL di produzione.

Stack tecnologico

- AWS CDK
- CodeBuild
- CodeCommit

- CodePipeline
- Amazon ECR
- Amazon ECS
- Amazon VPC

Automazione e scalabilità

Puoi utilizzare l'approccio di questo modello per creare pipeline per microservizi distribuiti in uno stack AWS condiviso. CloudFormation L'automazione può creare più di un cluster Amazon ECS in ogni VPC e anche creare pipeline per microservizi distribuiti in un cluster Amazon ECS condiviso. Tuttavia, ciò richiede che tu fornisca nuove informazioni sulle risorse come input per lo stack della pipeline.

Tools (Strumenti)

- AWS CDK — AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software per definire l'infrastruttura cloud in codice e fornirla tramite AWS. CloudFormation
- AWS CodeBuild: AWS CodeBuild è un servizio di build completamente gestito nel cloud. CodeBuild compila il codice sorgente, esegue test unitari e produce artefatti pronti per la distribuzione.
- AWS CodeCommit: AWS CodeCommit è un servizio di controllo delle versioni che consente di archiviare e gestire in modo privato repository Git nel cloud AWS. CodeCommit elimina la necessità di gestire il proprio sistema di controllo del codice sorgente o di preoccuparsi di scalarne l'infrastruttura.
- AWS CodePipeline: AWS CodePipeline è un servizio di distribuzione continua che puoi utilizzare per modellare, visualizzare e automatizzare i passaggi necessari per rilasciare il tuo software. Puoi modellare e configurare rapidamente le diverse fasi di un processo di rilascio del software. CodePipeline automatizza i passaggi necessari per rilasciare continuamente le modifiche al software.
- Amazon ECS — Amazon Elastic Container Service (Amazon ECS) è un servizio di gestione dei container veloce e altamente scalabile che viene utilizzato per eseguire, arrestare e gestire i container su un cluster. Puoi eseguire le tue attività e i tuoi servizi su un'infrastruttura serverless gestita da AWS Fargate. In alternativa, per un maggiore controllo sulla tua infrastruttura, puoi eseguire attività e servizi su un cluster di istanze Amazon Elastic Compute Cloud (Amazon EC2) da te gestite.
- Docker: Docker aiuta gli sviluppatori a imballare, spedire ed eseguire qualsiasi applicazione come contenitore leggero, portatile e autosufficiente.

Codice

Il codice per questo pattern è disponibile nei `starter-code.zip` file `cicdstarter.zip` and (allegati).

Epiche

Configurare l'ambiente

Operazione	Description	Competenze richieste
Configura la directory di lavoro per AWS CDK.	1. Crea una directory denominata `cicdproject` sul tuo computer locale. 2. Scaricate il `cicdstarter.zip` file (allegato) nella `cicdproject` cartella e decomprimetelo. Questo crea una cartella denominata`cicdstarter` . 3. Esegui il comando `cd <user-home>/cicdproject/cicdstarter` . 4. Configura l'ambiente virtuale Python eseguendo il `python3 -m venv .venv` comando. 5. Esegui il comando `source ./.venv/bin/activate` . 6. Configura il tuo ambiente AWS eseguendo il `aws configure` comando o utilizzando le seguenti variabili di ambiente:	AWS DevOps, infrastruttura cloud

Operazione	Description	Competenze richieste
	• `AWS_ACCESS_KEY_ID` • `AWS_SECRET_ACCESS_KEY` • `AWS_DEFAULT_REGION`	

Crea l'infrastruttura condivisa

Operazione	Description	Competenze richieste
Crea l'infrastruttura condivisa.	1. Nella tua directory di lavoro, esegui il `cd cicdvpcecs` comando. 2. Esegui il `pip3 install -r requirements.txt` comando per installare tutte le dipendenze Python richieste 3. Esegui `cdk bootstrap command` per impostare l'ambiente AWS per il CDK AWS. 4. Esegui il comando `cdk synth --context aws_account=<aws_account_ID> --context aws_region=<aws-region> .` 5. Esegui il comando `cdk deploy --context aws_account=<aws_account_ID> --`	AWS DevOps, infrastruttura cloud

Operazione	Description	Competenze richieste
	`context aws_regio n=<aws-region>` . 6. Lo CloudFormation stack AWS crea la seguente infrastruttura: • Un VPC non di produzion e denominato `cicd-vpc-ecs/cicd-vpc-nonprod` • Un VPC di produzione denominato `cicd-vpc-ecs/cicd-vpc-prod` • Un cluster Amazon ECS non di produzione denominato `cicd-ecs-nonprod` • Un cluster Amazon ECS di produzione denominato `cicd-ecs-prod`	
Monitora lo CloudFormation stack AWS.	1. Accedi alla Console di gestione AWS, apri la CloudFormation console AWS e scegli lo `cicd-vpc-ecs` stack dall'elen co. 2. Nel riquadro dei dettagli dello stack, scegli la scheda Eventi e monitora lo stato di avanzamento della creazione dello stack.	AWS DevOps, infrastruttura cloud

Operazione	Description	Competenze richieste
Testa lo CloudFormation stack AWS.	1. Dopo aver creato lo CloudFormation stack `cicd-vpc-ecs` AWS, assicurati che gli `cicd-vpc-ecs/cicd-vpc-nonprod` e `cicd-vpc-ecs/cicd-vpc-prod` VPCs siano stati creati. 2. Assicurati che i cluster `cicd-ecs-nonprod` e `cicd-ecs-prod` Amazon ECS siano creati. **Important** Assicurati di registrare il IDs per i due VPCs e il gruppo di sicurezza IDs per i gruppi di sicurezza predefiniti in entrambi. VPCs	AWS DevOps, infrastruttura cloud

Crea una CI/CD pipeline per un microservizio

Operazione	Description	Competenze richieste
Crea l'infrastruttura per il microservizio.	1. Assegna un nome al tuo microservizio. Ad esempio, questo modello utilizza `myservice1` come nome del microservizio.	AWS DevOps, infrastruttura cloud

Operazione	Description	Competenze richieste
	2. Nella tua directory di lavoro esegui il `cd <working-directory>/cdkpipeline` comando. 3. Esegui il comando `pip3 install -r requirements.txt .` 4. Esegui il `cdk synth` comando completo disponibile nella sezione Informazioni aggiuntive di questo modello. 5. Esegui il `cdk deploy` comando completo disponibile nella sezione Informazioni aggiuntive di questo modello. **Note** È inoltre possibile fornire i valori per entrambi i comandi utilizzando il `cdk.json` file nella directory.	

Operazione	Description	Competenze richieste
Monitora lo CloudFormation stack AWS.	Apri la CloudFormation console AWS e monitora l'avanzamento dello `myservice1-cicd-stack` stack. Alla fine, lo stato cambia in. *`CREATE_COMPLETE`*	AWS DevOps, infrastruttura cloud

Operazione	Description	Competenze richieste
Testa lo CloudFormation stack AWS.	1. Sulla CodeCommit console AWS, verifica che `myservice1` esista un repository denominato e contenga il codice di avvio. 2. Sulla CodeBuild console AWS, verifica che `myservice1` esista un progetto di build denominato. 3. Sulla console Amazon ECR, verifica che esista un repository Amazon ECR denominato. `myservice1` 4. Sulla console Amazon ECS, verifica che un servizio Fargate `myservice1` denominato esista sia in un cluster Amazon ECS non di produzione che in uno di produzione. 5. Sulla console Amazon Elastic Compute Cloud (Amazon EC2), verifica che siano stati creati gli Application Load Balancer non di produzione e di produzione. Registra i nomi DNS di. ALBs 6. Sulla CodePipeline console AWS, verifica che `myservice1` esista una pipeline denominat	

Operazione	Description	Competenze richieste
	a. Deve avere`Source`, `BuildDeploy-NonProd` , e `Deploy-Prod` fasi. La pipeline dovrebbe inoltre avere uno `in progress` status. 7. Monitora la pipeline fino al completamento di tutte le fasi. 8. Approvala manualmente per la produzione. 9. In una finestra del browser, inserisci i nomi DNS di. ALBs 10. L'applicazione dovrebbe essere visualizzata `Hello World` nelle aree di non produzione e produzione. URLs	

Operazione	Description	Competenze richieste
Usa la pipeline.	1. Apri il CodeCommit repository che hai creato in precedenza e apri il `index.js` file. 2. Sostituisci `Hello World` con `Hello CI/CD`. 3. Salva e conferma le modifiche nel ramo principal e. 4. Verificate che la pipeline abbia inizio e che la modifica attraversi le `Build` fasi `Deploy-NonProd` e `eDeploy-Prod` . 5. Approvate manualmente la produzione. 6. Ora URLs dovrebbero essere visualizzate sia la produzione che la non produzione. *`Hello CICD`*	AWS DevOps, infrastruttura cloud
Ripeti questa epopea per ogni microservizio.	Ripeti le attività di questa epopea per creare una CI/CD pipeline per ciascuno dei tuoi microservizi.	AWS DevOps, infrastruttura cloud

Risorse correlate

- Usare Python con AWS CDK
- Guida di riferimento in Python per AWS CDK
- Creazione di un servizio AWS Fargate utilizzando la CDK AWS

Informazioni aggiuntive

Comando **`cdk synth`**

```
cdk synth --context aws_account=<aws_account_number> --context
 aws_region=<aws_region> --context vpc_nonprod_id=<id_of_non_production
 VPC> --context vpc_prod_id=<id_of_production_VPC> --context
 ecssg_nonprod_id=< default_security_group_id_of_non-production_VPC>
 --context ecssg_prod_id=<default_security_group_id_of_production_VPC>
 --context code_commit_s3_bucket_for_code=<S3 bucket name> --context
 code_commit_s3_object_key_for_code=<Object_key_of_starter_code> --context
 microservice_name=<name_of_microservice>
```

`cdk deploy` command

```
cdk deploy --context aws_account=<aws_account_number> --context
 aws_region=<aws_region> --context vpc_nonprod_id=<id_of_non_production_VPC>
 --context vpc_prod_id=<id_of_production_VPC> --context ecssg_nonprod_id=<
 default_security_group_id_of_non-production_VPC> --context
 ecssg_prod_id=<default_security_group_id_of_production_VPC> --
context code_commit_s3_bucket_for_code=<S3 bucket name> --context
 code_commit_s3_object_key_for_code=<Object_key_of_starter_code> --context
 microservice_name=<name_of_microservice>
```

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Crea e invia immagini Docker ad Amazon ECR utilizzando GitHub Actions e Terraform

Ruchika Modi, Amazon Web Services

Riepilogo

Questo modello spiega come creare GitHub flussi di lavoro riutilizzabili per creare il tuo Dockerfile e inviare l'immagine risultante ad Amazon Elastic Container Registry (Amazon ECR). Il pattern automatizza il processo di creazione dei tuoi Dockerfile utilizzando Terraform e Actions. GitHub Ciò riduce al minimo la possibilità di errori umani e riduce notevolmente i tempi di implementazione.

Un'azione GitHub push sul ramo principale del GitHub repository avvia la distribuzione delle risorse. Il flusso di lavoro crea un repository Amazon ECR unico basato sulla combinazione del nome dell' GitHub organizzazione e del repository. Quindi invia l'immagine Dockerfile al repository Amazon ECR.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un GitHub account attivo.
- Un GitHub repository.
- Terraform versione 1 o successiva installata e configurata.
- Un bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) per il backend Terraform.
- Una tabella Amazon DynamoDB per lo state lock e la coerenza di Terraform. La tabella deve avere una chiave di partizione denominata `LockID` con un tipo di. `String` Se non è configurato, il blocco dello stato sarà disabilitato.
- Un ruolo AWS Identity and Access Management (IAM) con le autorizzazioni per configurare il backend Amazon S3 per Terraform. Per le istruzioni di configurazione, consulta la documentazione di Terraform.

Limitazioni

Questo codice riutilizzabile è stato testato solo con GitHub Actions.

Architecture

Stack tecnologico Target

- Repository Amazon ECR
- GitHub Azioni
- Terraform

Architettura di destinazione

Il diagramma illustra quanto segue:

1. Un utente aggiunge un modello Dockerfile e Terraform al repository. GitHub

2. Queste aggiunte avviano un flusso di lavoro Actions. GitHub

3. Il flusso di lavoro verifica se esiste un repository Amazon ECR. In caso contrario, crea il repository in base all' GitHub organizzazione e al nome del repository.

4. Il flusso di lavoro crea il Dockerfile e invia l'immagine al repository Amazon ECR.

Tools (Strumenti)

Servizi Amazon

- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro dei container gestito sicuro, scalabile e affidabile.

Altri strumenti

- GitHub Actions è integrato nella GitHub piattaforma per aiutarti a creare, condividere ed eseguire flussi di lavoro all'interno dei tuoi GitHub repository. Puoi utilizzare GitHub Actions per automatizzare attività come la creazione, il test e la distribuzione del codice.
- Terraform è uno strumento Infrastructure as Code (IaC) HashiCorp che ti aiuta a creare e gestire infrastrutture cloud e locali.

Archivio di codici

Il codice per questo pattern è disponibile nel repository GitHub Docker ECR Actions Workflow.

- Quando crei GitHub Actions, i file del flusso di lavoro Docker vengono salvati nella `/.github/workflows/` cartella di questo repository. Il flusso di lavoro per questa soluzione si trova nel file workflow.yaml.
- La `e2e-test` cartella fornisce un Dockerfile di esempio per riferimento e test.

Best practice

- Per le migliori pratiche per la scrittura di Dockerfile, consulta la documentazione Docker.
- Usa un endpoint VPC per Amazon ECR. Gli endpoint VPC sono basati su AWS PrivateLink, una tecnologia che consente di accedere in modo privato ad Amazon APIs ECR tramite indirizzi IP privati. Per le attività di Amazon ECS che utilizzano il tipo di avvio Fargate, l'endpoint VPC consente all'attività di estrarre immagini private da Amazon ECR senza assegnare un indirizzo IP pubblico all'attività.

Epiche

Configura il provider e l'archivio OIDC GitHub

Operazione	Description	Competenze richieste
Configura OpenID Connect.	Crea un provider OpenID Connect (OIDC). Utilizzerai il provider nella policy di fiducia per il ruolo IAM utilizzato in questa azione. Per istruzioni, consulta Configurazione di OpenID Connect in Amazon Web Services GitHub nella documentazione.	Amministratore AWS, AWS DevOps, AWS generale
Clona il GitHub repository.	Clona il repository GitHub Docker ECR Actions Workflow nella tua cartella locale: `$git clone https://github.com/aws-samp`	DevOps ingegnere

Operazione	Description	Competenze richieste
	`les/docker-ecr-act ions-workflow`	

Personalizza il flusso di lavoro GitHub riutilizzabile e distribuisci l'immagine Docker

Operazione	Description	Competenze richieste
Personalizza l'evento che avvia il flusso di lavoro Docker.	Il flusso di lavoro per questa soluzione è in workflow.yaml. Questo script è attualmen te configurato per distribuire risorse quando riceve l'evento. `workflow_dispatch` È possibile personalizzare questa configurazione modificando l'evento in un altro flusso di lavoro principal e `workflow_call` e richiamando il flusso di lavoro da un altro flusso di lavoro principale.	DevOps ingegnere
Personalizza il flusso di lavoro.	Il file workflow.yaml è configurato per creare un flusso di lavoro dinamico e riutilizzabile. GitHub Puoi modificare questo file per personalizzare la configura zione predefinita oppure puoi passare i valori di input dalla console GitHub Actions se utilizzi l'evento per avviare la `workflow_dispatch` distribuzione manualmente.	DevOps ingegnere

Operazione	Description	Competenze richieste

	• Assicurati di specificare l'ID dell'account AWS e la regione di destinazione corretti.
	• Crea una policy sul ciclo di vita di Amazon ECR (vedi policy di esempio) e aggiorna il percorso predefinito (`e2e-test/policy.json`) di conseguenza.
	• Il file di workflow richiede due ruoli IAM come input:
	  • Un ruolo IAM che dispone delle autorizzazioni per configurare il backend Amazon S3 per Terraform (vedi la sezione Prerequisiti). Puoi aggiornare il nome del ruolo predefinito in. `workload-assumable-role yaml`file di conseguenza.
	  • Un ruolo IAM con autorizzazioni di accesso GitHub. Questo ruolo viene utilizzato anche nella policy di Amazon ECR per limitare le operazioni di Amazon ECR. Per ulteriori informazioni, consulta il file data.tf.	

Operazione	Description	Competenze richieste
Implementa i modelli Terraform.	Il flusso di lavoro distribuisce automaticamente i modelli Terraform che creano il repository Amazon ECR, in base all' GitHub evento configurato. Questi modelli sono disponibili come `.tf` file nella radice del repository Github.	AWS DevOps, DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
Problemi o errori durante la configurazione di Amazon S3 e DynamoDB come backend remoto Terraform.	Segui le istruzioni nella documentazione di Terraform per configurare le autorizzazioni richieste sulle risorse Amazon S3 e DynamoDB per la configurazione del backend remoto.
Impossibile eseguire o avviare il flusso di lavoro con l'evento. `workflow_dispatch`	Il flusso di lavoro configurato per l'impleme ntazione dall'`workflow_dispatch` evento funzionerà solo se è configurato anche nel ramo principale.

Risorse correlate

- Riutilizzo dei flussi di lavoro (documentazione) GitHub
- Attivazione di un flusso di lavoro (documentazione) GitHub

Crea e testa app iOS con AWS CodeCommit, AWS e CodePipeline AWS Device Farm

Abdullahi Olaoye, Amazon Web Services

Riepilogo

Questo modello delinea i passaggi per creare una pipeline di integrazione e distribuzione continua (CI/CD) che utilizzi AWS per creare e CodePipeline testare applicazioni iOS su dispositivi reali su AWS. Il modello utilizza AWS CodeCommit per archiviare il codice dell'applicazione, lo strumento open source Jenkins per creare l'applicazione iOS e AWS Device Farm per testare l'applicazione costruita su dispositivi reali. Queste tre fasi sono orchestrate insieme in una pipeline utilizzando AWS. CodePipeline

Questo modello si basa sul post Creazione e test di app iOS e iPadOS con AWS DevOps e servizi mobili sul DevOps blog AWS. Per istruzioni dettagliate, consulta il post del blog.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un account per sviluppatori Apple
- Crea server (macOS)
- Xcode versione 11.3 (installata e configurata sul server di build)
- AWS Command Line Interface (AWS CLI) installata e configurata sulla workstation
- Conoscenza di base di Git

Limitazioni

- Il server di creazione dell'applicazione deve eseguire macOS.
- Il server di compilazione deve avere un indirizzo IP pubblico, in modo da CodePipeline potervi connettere in remoto per avviare le build.

Architecture

Stack tecnologico di origine

- Un processo di creazione di applicazioni iOS in locale che prevede l'utilizzo di un simulatore o di un test manuale su dispositivi fisici

Stack tecnologico Target

- Un CodeCommit repository AWS per l'archiviazione del codice sorgente dell'applicazione
- Un server Jenkins per la creazione di applicazioni utilizzando Xcode
- Un pool di dispositivi AWS Device Farm per testare applicazioni su dispositivi reali

Architettura Target

Quando un utente esegue il commit delle modifiche al repository di origine, la pipeline (AWS CodePipeline) recupera il codice dal repository di origine, avvia una build Jenkins e passa il codice dell'applicazione a Jenkins. Dopo la compilazione, la pipeline recupera l'elemento della build e avvia un job di AWS Device Farm per testare l'applicazione su un pool di dispositivi.

Tools (Strumenti)

- AWS CodePipeline è un servizio di distribuzione continua completamente gestito che ti aiuta ad automatizzare le pipeline di rilascio per aggiornamenti rapidi e affidabili di applicazioni e infrastrutture. CodePipeline automatizza le fasi di compilazione, test e distribuzione del processo di rilascio ogni volta che viene apportata una modifica al codice, in base al modello di rilascio definito.
- AWS CodeCommit è un servizio di controllo del codice sorgente completamente gestito che ospita repository sicuri basati su Git. Permette ai team di collaborare facilmente sul codice in un ecosistema sicuro e altamente scalabile. CodeCommit elimina la necessità di gestire il proprio sistema di controllo del codice sorgente o di preoccuparsi di scalarne l'infrastruttura.
- AWS Device Farm è un servizio di test delle applicazioni che ti consente di migliorare la qualità delle tue app web e mobili testandole su un'ampia gamma di browser desktop e dispositivi mobili reali, senza dover fornire e gestire alcuna infrastruttura di test.
- Jenkins è un server di automazione open source che consente agli sviluppatori di creare, testare e distribuire il proprio software.

Epiche

Configura l'ambiente di compilazione

Operazione	Description	Competenze richieste
Installa Jenkins sul server di build che esegue macOS.	Jenkins verrà utilizzato per creare l'applicazione, quindi devi prima installarlo sul server di compilazione. Per ottenere istruzioni dettagliate per questa e le attività successive, consulta il post sul blog di AWS Creazione e test di app iOS e iPadOS con AWS DevOps e servizi mobili e altre risorse nella sezione Risorse correlate alla fine di questo schema.	DevOps
Configura Jenkins.	Segui le istruzioni sullo schermo per configurare Jenkins.	DevOps
Installa il CodePipeline plugin AWS per Jenkins.	Questo plugin deve essere installato sul server Jenkins per consentire a Jenkins di interagire con il servizio CodePipeline AWS.	DevOps
Crea un progetto Jenkins freestyle.	In Jenkins, crea un progetto freestyle. Configura il progetto per specificare i trigger e altre opzioni di configurazione della build.	DevOps

Configurazione di AWS Device Farm

Operazione	Description	Competenze richieste
Crea un progetto Device Farm.	Apri la console AWS Device Farm. Crea un progetto e un pool di dispositivi per i test. Per istruzioni, consulta il post del blog.	Developer

Configura il repository dei sorgenti

Operazione	Description	Competenze richieste
Crea un CodeCommit repository.	Crea un repository in cui verrà archiviato il codice sorgente.	DevOps
Salva il codice dell'appl icazione nel repository.	Connect al CodeCommit repository che hai creato. Invia il codice dal tuo computer locale al repository.	DevOps

Configura la pipeline

Operazione	Description	Competenze richieste
Crea una pipeline in AWS CodePipeline.	Apri la CodePipeline console AWS e crea una pipeline. La pipeline orchestra tutte le fasi del processo. CI/CD Per istruzioni, consulta il post sul blog AWS Creazione e test di app iOS e iPadOS con AWS DevOps e servizi mobili.	DevOps

Operazione	Description	Competenze richieste
Aggiungi una fase di test alla pipeline.	Per aggiungere una fase di test e integrarla con AWS Device Farm, modifica la pipeline.	DevOps
Avvia la pipeline.	Per avviare la pipeline e il CI/CD processo, scegliete Release change.	DevOps

Visualizza i risultati dei test applicativi

Operazione	Description	Competenze richieste
Rivedi i risultati dei test.	Nella console AWS Device Farm, seleziona il progetto che hai creato ed esamina i risultati dei test. La console mostrerà i dettagli di ogni test.	Developer

Risorse correlate

Step-by-step istruzioni per questo modello

- Creazione e test di app iOS e iPadOS con AWS DevOps e servizi mobili (post DevOps sul blog AWS)

Configurazione di AWS Device Farm

- Console AWS Device Farm

Configura il repository di origine

- Crea un CodeCommit repository AWS
- Connect a un CodeCommit repository AWS

Configura la pipeline

- CodePipeline Console AWS

Altre risorse

- CodePipeline Documentazione AWS
- CodeCommit Documentazione AWS
- Documentazione di AWS Device Farm
- Documentazione Jenkins
- Installazione di Jenkins su macOS
- CodePipeline Plugin AWS per Jenkins
- Installazione di Xcode
- Installazione e configurazione dell'interfaccia a riga di comando di AWS
- Documentazione Git

Configura l'autenticazione TLS reciproca per le applicazioni in esecuzione su Amazon EKS

Mahendra Revanasiddappa, Amazon Web Services

Riepilogo

Il Mutual Transport Layer Security (TLS) basato su certificati è un componente TLS opzionale che fornisce l'autenticazione peer bidirezionale tra server e client. Con Mutual TLS, i client devono fornire un certificato X.509 durante il processo di negoziazione della sessione. Il server utilizza questo certificato per identificare e autenticare il client.

Il Mutual TLS è un requisito comune per le applicazioni Internet of Things (IoT) e può essere utilizzato per business-to-business applicazioni o standard come l'Open Banking.

Questo modello descrive come configurare il TLS reciproco per le applicazioni in esecuzione su un cluster Amazon Elastic Kubernetes Service (Amazon EKS) utilizzando un controller di ingresso NGINX. Puoi abilitare le funzionalità TLS reciproche integrate per il controller di ingresso NGINX annotando la risorsa di ingresso. Per ulteriori informazioni sulle annotazioni TLS reciproche sui controller NGINX, consulta Autenticazione dei certificati client nella documentazione di Kubernetes.

> **Important**
>
> Questo modello utilizza certificati autofirmati. Si consiglia di utilizzare questo modello solo con i cluster di test e non negli ambienti di produzione. Se desideri utilizzare questo modello in un ambiente di produzione, puoi utilizzare AWS Private Certificate Authority (AWS Private CA) o lo standard esistente di infrastruttura a chiave pubblica (PKI) per emettere certificati privati.

Prerequisiti e limitazioni

Prerequisiti

- Un account Amazon Web Services (AWS) attivo.
- Un cluster Amazon EKS esistente.
- AWS Command Line Interface (AWS CLI) versione 1.7 o successiva, installata e configurata su macOS, Linux o Windows.

- L'utilità da riga di comando kubectl, installata e configurata per accedere al cluster Amazon EKS. Per ulteriori informazioni su questo argomento, consulta Installazione di kubectl nella documentazione di Amazon EKS.
- Un nome DNS (Domain Name System) esistente per testare l'applicazione.

Limitazioni

- Questo modello utilizza certificati autofirmati. Si consiglia di utilizzare questo modello solo con i cluster di test e non negli ambienti di produzione.

Architecture

Stack tecnologico

- Amazon EKS
- Amazon Route 53
- Kubectl

Tools (Strumenti)

- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire Kubernetes su AWS senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- Amazon Route 53 è un servizio Web DNS altamente scalabile e disponibile.
- Kubectl è un'utilità da riga di comando che usi per interagire con un cluster Amazon EKS.

Epiche

Genera i certificati autofirmati

Operazione	Description	Competenze richieste
Genera la chiave CA e il certificato.	Genera la chiave e il certifica to dell'autorità di certificazione (CA) eseguendo il comando seguente.	DevOps ingegnere

Operazione	Description	Competenze richieste
	`openssl req -x509 -sha256 -newkey rsa:4096 -keyout ca.key -out ca.crt -days 356 -nodes -subj '/CN=Test Cert Authority'`	
Genera la chiave e il certificato del server e firma con il certificato CA.	Genera la chiave e il certificato del server e firma con il certificato CA eseguendo il comando seguente. `openssl req -new -newkey rsa:4096 -keyout server.key -out server.csr -nodes -subj '/CN= <your_domain_name> ' && openssl x509 -req -sha256 -days 365 -in server.csr -CA ca.crt -CAkey ca.key -set_serial 01 -out server.crt` **Important** Assicurati di sostituire lo `<your_domain_name>` con il nome di dominio esistente.	DevOps ingegnere

Operazione	Description	Competenze richieste
Genera la chiave client e il certificato e firma con il certificato CA.	Genera la chiave client e il certificato e firma con il certificato CA eseguendo il comando seguente. `openssl req -new -newkey rsa:4096 -keyout client.key -out client.csr -nodes -subj '/CN=Test' && openssl x509 -req -sha256 -days 365 -in client.csr -CA ca.crt -CAkey ca.key -set_serial 02 -out client.crt`	DevOps ingegnere

Implementa il controller di ingresso NGINX

Operazione	Description	Competenze richieste
Implementa il controller di ingresso NGINX nel tuo cluster Amazon EKS.	Implementa il controller di ingresso NGINX utilizzando il seguente comando. `kubectl apply -f https://raw.githubusercontent.com/kubernetes/ingress-nginx/controller-v1.7.0/deploy/static/provider/aws/deploy.yaml`	DevOps ingegnere
Verifica che il servizio di controllo di ingresso NGINX sia in esecuzione.	Verifica che il servizio di controllo di ingresso NGINX	DevOps ingegnere

Operazione	Description	Competenze richieste
	sia in esecuzione utilizzando il seguente comando. `kubectl get svc -n ingress-nginx` **Important** Assicurati che l'indirizzo del campo di servizio contenga il nome di dominio del Network Load Balancer.	

Crea uno spazio dei nomi nel cluster Amazon EKS per testare il TLS reciproco

Operazione	Description	Competenze richieste
Crea uno spazio dei nomi nel cluster Amazon EKS.	Crea uno spazio dei nomi chiamato `mtls` nel tuo cluster Amazon EKS eseguendo il comando seguente. `kubectl create ns mtls` Questo implementa l'applicazione di esempio per testare il TLS reciproco.	DevOps ingegnere

Crea la distribuzione e il servizio per l'applicazione di esempio

Operazione	Description	Competenze richieste

Crea la distribuzione e il servizio Kubernetes nello spazio dei nomi mtls.	Crea un file denominato `mtls.yaml` . Incolla il codice seguente nel file.
	
	`kind: Deployment`
	`apiVersion: apps/v1`
	`metadata:`
	`  name: mtls-app`
	`  labels:`
	`  app: mtls`
	`spec:`
	`  replicas: 1`
	`  selector:`
	`  matchLabels:`
	`  app: mtls`
	`  template:`
	`  metadata:`
	`  labels:`
	`  app: mtls`
	`  spec:`
	`  containers:`
	`  - name: mtls-app`
	`  image: hashicorp/http-echo`
	`  args:`
	`  - "-text=mTLS is working"`
	
	`---`
	
	`kind: Service`
	`apiVersion: v1`
	`metadata:`
	`  name: mtls-service`
	`spec:`
	`  selector:`
	`  app: mtls`
		DevOps ingegnere

Operazione	Description	Competenze richieste

	`ports:`
	`- port: 5678 #`
	`Default port for image`
	
	Crea la distribuzione e il servizio Kubernetes nello spazio dei nomi eseguendo il comando seguente. `mtls`
	
	`kubectl create -f mtls.yaml -n mtls`	

Verifica che la distribuzione Kubernetes sia stata creata.	Esegui il comando seguente per verificare che la distribuzione sia stata creata e che un pod sia disponibile. `kubectl get deploy -n mtls`	DevOps ingegnere
Verifica che il servizio Kubernetes sia stato creato.	Verifica che il servizio Kubernetes sia stato creato eseguendo il comando seguente. `kubectl get service -n mtls`	DevOps ingegnere

Crea un segreto nel namespace mtls

Operazione	Description	Competenze richieste
Crea un segreto per la risorsa in ingresso.	Esegui il seguente comando per creare un segreto per il controller di ingresso NGINX	DevOps ingegnere

Operazione	Description	Competenze richieste
	utilizzando i certificati che hai creato in precedenza. `kubectl create secret generic mtls-certs --from-file=tls.crt=server.crt --from-file=tls.key=server.key --from-file=ca.crt=ca.crt -n mtls` Il tuo segreto ha un certificato server che consente al client di identificare il server e un certificato CA per il server per verificare i certificati client.	

Crea la risorsa di ingresso nello spazio dei nomi mtls

Operazione	Description	Competenze richieste

Crea la risorsa di ingresso nello spazio dei nomi mtls.	Crea un file denominato `ingress.yaml` . Incolla il seguente codice nel file (sostituiscilo `<your_domain_name>` con il tuo nome di dominio esistente).
	
	`apiVersion: networking.k8s.io/v1`
	`kind: Ingress`
	`metadata:`
	`  annotations:`
	`  nginx.ingress.kubernetes.io/auth-tls-verify-client: "on"`
		DevOps ingegnere

Operazione	Description	Competenze richieste

	<pre> nginx.ingress.kube
	rnetes.io/auth-tls-
	secret: mtls/mtls-certs
	  name: mtls-ingress
	spec:
	  ingressClassName:
	 nginx
	  rules:
	  - host: "*.<your_
	domain_name>"
	    http:
	      paths:
	      - path: /
	        pathType: Prefix
	        backend:
	          service:
	            name: mtls-
	service
	            port:
	              number:
	 5678
	  tls:
	  - hosts:
	    - "*.<your_
	domain_name>"
	    secretName: mtls-
	certs</pre>
	Crea la risorsa di ingresso nel `mtls` namespace eseguendo il comando seguente.
	<pre>kubectl create -f
	 ingress.yaml -n mtls</pre>
	Ciò significa che il controller di ingresso NGINX può indirizza re il traffico verso l'applica zione di esempio.	

Operazione	Description	Competenze richieste
Verifica che la risorsa in ingresso sia stata creata.	Verificate che la risorsa di ingresso sia stata creata eseguendo il comando seguente. `kubectl get ing -n mtls` **Important** Assicurati che l'indirizzo della risorsa in ingresso mostri il load balancer creato per il controller di ingresso NGINX.	DevOps ingegnere

Configura il DNS per indirizzare il nome host verso il sistema di bilanciamento del carico

Operazione	Description	Competenze richieste
Crea un record CNAME che punti al load balancer per il controller di ingresso NGINX.	Accedi alla Console di gestione AWS, apri la console Amazon Route 53 e crea un record Canonical Name (CNAME) che punti `mtls.<your_domain_name>` al load balancer per il controller di ingresso NGINX. Per ulteriori informazioni, vedere Creazione di record utilizzando la console Route	DevOps ingegnere

Operazione	Description	Competenze richieste
	53 nella documentazione di Route 53.	

Eseguire il test dell'applicazione

Operazione	Description	Competenze richieste
Prova la configurazione TLS reciproca senza certificati.	Eseguire il seguente comando seguente. `curl -k https://m tls.<your_domain_n ame>` Dovresti ricevere la risposta di errore «400 Nessun certificato SSL richiesto è stato inviato».	DevOps ingegnere
Testa la configurazione TLS reciproca con i certificati.	Eseguire il seguente comando seguente. `curl -k https://m tls.<your_domain_n ame> --cert client.crt --key client.key` Dovresti ricevere la risposta «mTLS funziona».	DevOps ingegnere

Risorse correlate

- Creazione di record utilizzando la console Amazon Route 53
- Utilizzo di un Network Load Balancer con il controller di ingresso NGINX su Amazon EKS
- Autenticazione tramite certificato client

Automatizza la creazione di risorse Amazon WorkSpaces Applications utilizzando AWS CloudFormation

Ram Kandaswamy, Amazon Web Services

Riepilogo

Questo modello fornisce esempi di codice e passaggi per automatizzare la creazione di risorse Amazon WorkSpaces Applications Cloud AWS utilizzando un AWS CloudFormationmodello. Il modello mostra come utilizzare uno CloudFormation stack per automatizzare la creazione delle risorse delle WorkSpaces applicazioni Applications, tra cui un generatore di immagini, un'immagine, un'istanza di flotta e uno stack. È possibile trasmettere in streaming l'applicazione WorkSpaces Applications agli utenti finali su un browser HTML5 conforme utilizzando la modalità di distribuzione desktop o dell'applicazione.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un'accettazione dei termini e WorkSpaces delle condizioni delle Applicazioni
- Conoscenza di base delle risorse delle WorkSpaces Applicazioni, come flotte e stack e generatori di immagini

Limitazioni

- Non è possibile modificare il ruolo AWS Identity and Access Management(IAM) associato a un'istanza WorkSpaces Applications dopo la creazione di tale istanza.
- Non è possibile modificare le proprietà (come la sottorete o il gruppo di sicurezza) sull'istanza WorkSpaces Applications image builder dopo la creazione del generatore di immagini.

Architecture

Il diagramma seguente mostra come automatizzare la creazione di risorse delle WorkSpaces applicazioni utilizzando un modello. CloudFormation

Il diagramma mostra il flusso di lavoro seguente:

1. È possibile creare un CloudFormation modello basato sul codice YAML nella sezione Informazioni aggiuntive di questo modello.
2. Il CloudFormation modello crea uno stack di CloudFormation test.
   a. (Facoltativo) È possibile creare un'istanza di image builder utilizzando WorkSpaces Applicazioni.
   b. (Facoltativo) È possibile creare un'immagine Windows utilizzando un software personalizzato.
3. Lo CloudFormation stack crea un'istanza e uno stack del parco WorkSpaces applicazioni.
4. WorkSpaces Le risorse delle applicazioni vengono distribuite agli utenti finali su un HTML5 browser conforme.

Tools (Strumenti)

- Amazon WorkSpaces Applications è un servizio di streaming di applicazioni completamente gestito che ti fornisce accesso immediato alle tue applicazioni desktop da qualsiasi luogo. WorkSpaces Applications gestisce le AWS risorse necessarie per ospitare ed eseguire le applicazioni, si ridimensiona automaticamente e fornisce l'accesso agli utenti su richiesta.
- AWS CloudFormationti aiuta a modellare e configurare le AWS risorse, a fornirle in modo rapido e coerente e a gestirle per tutto il loro ciclo di vita. È possibile utilizzare un modello per descrivere le risorse e le relative dipendenze e lanciarle e configurarle insieme come uno stack, anziché gestire le risorse singolarmente. Puoi gestire ed eseguire il provisioning degli stack su più sistemi. Account AWS Regioni AWS

Best practice

- Configura correttamente l'accesso alla rete per i generatori di immagini: avvia i generatori di immagini nelle sottoreti del cloud privato virtuale (VPC) con un accesso Internet adeguato utilizzando un gateway NAT per l'accesso a Internet solo in uscita.

  Verifica la connettività di rete con le risorse richieste (come server delle applicazioni, database e server di licenza) prima di creare immagini. Verifica che le tabelle di routing VPC consentano le connessioni a tutte le risorse di rete necessarie. Per ulteriori informazioni, consulta l'accesso a Internet nella documentazione delle WorkSpaces applicazioni.
- Monitora la capacità della flotta rispetto alle quote di servizio in modo proattivo: WorkSpaces le quote per tipo e dimensione delle istanze delle applicazioni si intendono per, per. Account AWS Regione AWS Se disponi di più parchi istanze nella stessa regione che utilizzano lo stesso tipo

e dimensione di istanze, il numero totale di istanze in tutti i parchi istanze di quella regione deve essere inferiore o uguale alla quota applicabile. Per ulteriori informazioni, consulta Risoluzione dei problemi relativi alle flotte nella documentazione delle applicazioni. WorkSpaces

- Testa le applicazioni in modalità Image Builder Test prima dell'implementazione in flotta: convalida sempre le applicazioni in modalità Image Builder Test prima di creare immagini e distribuirle alle flotte. La modalità Test simula le autorizzazioni limitate di cui dispongono gli utenti finali sulle istanze del parco istanze. Per ulteriori informazioni, consulta Troubleshooting Image Builders nella documentazione delle applicazioni. WorkSpaces

Epiche

(Facoltativo) Crea un'immagine WorkSpaces delle applicazioni

Operazione	Description	Competenze richieste
Installa software personalizzato e crea un'immagine.	1. Avvia Image Builder e scegli i tipi di istanza, la sottorete, il gruppo di sicurezza pertinenti e così via. 2. Usa la modalità amministratore durante la connessione all'istanza. 3. Usa l'Image Assistant o uno PowerShell script per creare una nuova immagine Windows per il tuo software personalizzato. **Note** Prendi in considerazione l'utilizzo della AppLocker funzionalità Windows per	AWS DevOps, architetto cloud

Operazione	Description	Competenze richieste
	bloccare ulteriormente l'immagine.	

Implementa il modello CloudFormation

Operazione	Description	Competenze richieste
Aggiorna il CloudFormation modello.	1. Usa il codice nella sezione Informazioni aggiuntive di questo modello come file YAML. 2. Aggiorna il file YAML con i valori richiesti per i parametri del tuo ambiente.	Amministratore di sistema AWS, amministratore cloud, architetto cloud, General AWS, amministratore AWS
Crea uno CloudFormation stack utilizzando il modello.	1. Accedi a Console di gestione AWS e apri la CloudFormation console. 2. Nel pannello di navigazione, scegli Stacks. 3. Scegliere Create stack (Crea stack), quindi With new resources (standard) (Con nuove risorse (standard)). 4. Nella sezione Prerequisito: prepara il modello, scegli Il modello è pronto. 5. Nella sezione Specificare il modello, scegli Carica un file modello.	Proprietario dell'app, amministratore di sistema AWS, Windows Engineer

Operazione	Description	Competenze richieste
	6. Scegli il file, quindi scegli il CloudFormation modello aggiornato. 7. Completa il resto dei passaggi della procedura guidata per creare lo stack.	

Risoluzione dei problemi

Problema	Soluzione
Vari problemi	Per ulteriori informazioni, consulta Risoluzione dei problemi nella documentazione WorkSpaces delle applicazioni.

Risorse correlate

Riferimenti

- Inizia a usare Amazon WorkSpaces Applications: configura con applicazioni di esempio
- Crea una flotta e uno stack di WorkSpaces applicazioni Amazon

Tutorial e video

- Flusso di lavoro degli utenti di Amazon WorkSpaces Applications
- Come migrare un'app Windows Forms legacy su Amazon Applications WorkSpaces
- AWS re:Invent 2018: Distribuisci applicazioni desktop in modo sicuro con Amazon WorkSpaces Applications (01) BAP2

Informazioni aggiuntive

Il codice seguente è un esempio di CloudFormation modello che puoi utilizzare per creare automaticamente le risorse delle applicazioni. WorkSpaces

```
AWSTemplateFormatVersion: 2010-09-09
Parameters:
  SubnetIds:
    Type: 'List<AWS::EC2::Subnet::Id>'
  testSecurityGroup:
    Type: 'AWS::EC2::SecurityGroup::Id'
  ImageName:
    Type: String
Resources:

  AppStreamFleet:
    Type: 'AWS::AppStream::Fleet'
    Properties:
      ComputeCapacity:
        DesiredInstances: 5
      InstanceType: stream.standard.medium
      Name: appstream-test-fleet
      DisconnectTimeoutInSeconds: 1200
      FleetType: ON_DEMAND
      IdleDisconnectTimeoutInSeconds: 1200
      ImageName: !Ref ImageName
      MaxUserDurationInSeconds: 345600
      VpcConfig:
        SecurityGroupIds:
          - !Ref testSecurityGroup
        SubnetIds: !Ref SubnetIds
  AppStreamStack:
    Type: 'AWS::AppStream::Stack'
    Properties:
      Description: AppStream stack for test
      DisplayName: AppStream test Stack
      Name: appstream-test-stack
      StorageConnectors:
        - ConnectorType: HOMEFOLDERS
      UserSettings:
        - Action: CLIPBOARD_COPY_FROM_LOCAL_DEVICE
          Permission: ENABLED
        - Action: CLIPBOARD_COPY_TO_LOCAL_DEVICE
          Permission: ENABLED
        - Action: FILE_DOWNLOAD
          Permission: ENABLED
        - Action: PRINTING_TO_LOCAL_DEVICE
          Permission: ENABLED
```

```
  AppStreamFleetAssociation:
    Type: 'AWS::AppStream::StackFleetAssociation'
    Properties:
      FleetName: appstream-test-fleet
      StackName: appstream-test-stack
    DependsOn:
      - AppStreamFleet
      - AppStreamStack
```

Crea un parser di log personalizzato per Amazon ECS utilizzando un router di log Firelens

Varun Sharma, Amazon Web Services

Riepilogo

Firelens è un router di log per Amazon Elastic Container Service (Amazon ECS) e AWS Fargate. Puoi utilizzare Firelens per instradare i log dei container da Amazon ECS ad CloudWatch Amazon e ad altre destinazioni (ad esempio, Splunk o Sumo Logic). Firelens funziona con Fluentd o Fluent Bit come agente di registrazione, il che significa che puoi utilizzare i parametri di definizione delle attività di Amazon ECS per instradare i log.

Scegliendo di analizzare i log a livello di origine, puoi analizzare i dati di registrazione ed eseguire query per rispondere in modo più efficiente ed efficace ai problemi operativi. Poiché applicazioni diverse hanno modelli di registrazione diversi, è necessario utilizzare un parser personalizzato che struttura i log e faciliti la ricerca nella destinazione finale.

Questo modello utilizza un router di log Firelens con un parser personalizzato a cui inviare i log CloudWatch da un'applicazione Spring Boot di esempio in esecuzione su Amazon ECS. Puoi quindi utilizzare Amazon CloudWatch Logs Insights per filtrare i log in base a campi personalizzati generati dal parser personalizzato.

Prerequisiti e limitazioni

Prerequisiti

- Un account Amazon Web Services (AWS) attivo.
- AWS Command Line Interface (AWS CLI), installata e configurata sul computer locale.
- Docker, installato e configurato sul tuo computer locale.
- Un'applicazione containerizzata esistente basata su Spring Boot su Amazon Elastic Container Registry (Amazon ECR).

Architecture

Stack tecnologico

- CloudWatch
- Amazon ECR
- Amazon ECS
- Fargate
- Docker
- Fluent Bit

Tools (Strumenti)

- Amazon ECR — Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito da AWS sicuro, scalabile e affidabile.
- Amazon ECS — Amazon Elastic Container Service (Amazon ECS) è un servizio di gestione dei container veloce e altamente scalabile che semplifica l'esecuzione, l'arresto e la gestione dei container su un cluster.
- AWS Identity and Access Management (IAM): IAM è un servizio Web per controllare in modo sicuro l'accesso ai servizi AWS.
- AWS CLI — AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire con i servizi AWS utilizzando i comandi nella shell della riga di comando.
- Docker: Docker è una piattaforma aperta per lo sviluppo, la spedizione e l'esecuzione di applicazioni.

Codice

I seguenti file sono allegati a questo modello:

- `customFluentBit.zip`— Contiene i file per aggiungere l'analisi e le configurazioni personalizzate.
- `firelens_policy.json`— Contiene il documento di policy per creare una policy IAM.
- `Task.json`— Contiene una definizione di attività di esempio per Amazon ECS.

Epiche

Crea un'immagine Fluent Bit personalizzata

Operazione	Description	Competenze richieste
Crea un repository Amazon ECR.	Accedi alla Console di gestione AWS, apri la console Amazon ECR e crea un repository chiamato. `fluentbit_custom` Per ulteriori informazioni su questo argomento, consulta Creazione di un repository nella documentazione di Amazon ECR.	Amministratore di sistema, sviluppatore
Decomprimi il customFluentBit pacchetto.zip.	1. Scarica il `customFluentBit.zip` pacchetto (allegato) sul tuo computer locale. 2. Decomprimi nella `customFluentBit` directory eseguendo il seguente comando: `unzip -d customFluentBit.zip` 3. La directory contiene i seguenti file necessari per aggiungere l'analisi e le configurazioni personalizzate: • `parsers/springboot_parser.conf` —	

Operazione	Description	Competenze richieste
	Contiene la direttiva parser e definisce il modello di espressione regolare (regex) per il parser personalizzato. Puoi aggiungere il pattern regex per il tuo parser specifico. • `conf/parse_springboot.conf` — Contiene il filtro e la direttiva di servizio. • Il Dockerfile	

Operazione	Description	Competenze richieste
Crea l'immagine Docker personalizzata.	1. Cambiare la directory in `customFluentBit` . 2. Apri la console Amazon ECR, scegli il `fluentbit_custom` repository, quindi scegli Visualizza comandi push. 3. Carica il tuo progetto. 4. Una volta completato il caricamento, copia l'URL della build. Questo URL è obbligatorio quando crei un contenitore in Amazon ECS Per ulteriori informazioni su questo argomento, consulta Pushing a Docker image nella documentazione di Amazon ECR.	Amministratore di sistema, sviluppatore

Configura il cluster Amazon ECS

Operazione	Description	Competenze richieste
Crea un cluster Amazon ECS.	Crea un cluster Amazon ECS seguendo le istruzioni dalla sezione Modello solo per reti di rete di Creazione di un cluster nella documentazione di Amazon ECS.	Amministratore di sistema, sviluppatore

Operazione	Description	Competenze richieste
	Note Assicurati di scegliere Create VPC per creare un nuovo cloud privato virtuale (VPC) per il tuo cluster Amazon ECS.	

Configurare l'attività Amazon ECS

Operazione	Description	Competenze richieste
Configura il ruolo IAM di esecuzione delle attività di Amazon ECS.	Crea un ruolo IAM per l'esecuzione delle attività di Amazon ECS utilizzando la policy `AmazonECSTaskExecutionRolePolicy` gestita. Per ulteriori informazioni su questo argomento, consulta il ruolo IAM di esecuzione delle attività di Amazon ECS nella documentazione di Amazon ECS. **Note** Assicurati di registrare l'Amazon Resource Name (ARN) del ruolo IAM.	Amministratore di sistema, sviluppatore

Operazione	Description	Competenze richieste
Collega la policy IAM al ruolo IAM di esecuzione delle attività di Amazon ECS.	1. Crea una policy IAM utilizzando il documento di policy `firelens_policy.json` (allegato). Per ulteriori informazioni a riguardo, consulta Creazione di politiche nella scheda JSON nella documentazione IAM. 2. Collega questa policy al ruolo IAM di esecuzione delle attività di Amazon ECS creato in precedenza. Per ulteriori informazioni su questo argomento, consulta Aggiungere politiche IAM (AWS CLI) nella documentazione IAM.	Amministratore di sistema, sviluppatore

Operazione	Description	Competenze richieste
Imposta la definizione del task di Amazon ECS.	1. Aggiorna le seguenti sezioni nella definizione di attività `Task.json` di esempio (allegata): • Aggiorna l'`executionRoleArn` and `taskRoleArn` con l'ARN del ruolo IAM di esecuzione dell'attività • Aggiorna l'immagine `containerDefinitions` con l'immagine Fluent Bit Docker personalizzata che hai creato in precedenza • Aggiorna l'immagine `containerDefinitions` con il nome dell'immagine dell'applicazione 2. Apri la console Amazon ECS, scegli Definizioni attività, scegli Crea nuova definizione di attività, quindi scegli Fargate nella pagina Seleziona compatibilità. 3. Scegli Configura tramite Json, incolla il **`Task.json`** file aggiornato nell'area di testo, quindi scegli Salva. 4. Crea la definizione dell'attività.	Amministratore di sistema, sviluppatore

Operazione	Description	Competenze richieste
	Per ulteriori informazioni su questo argomento, consulta Creazione di una definizione di attività nella documentazione di Amazon ECS.	

Esegui l'attività Amazon ECS

Operazione	Description	Competenze richieste
Esegui l'attività Amazon ECS.	Sulla console Amazon ECS, scegli Clusters, scegli il cluster che hai creato in precedenza, quindi esegui l'attività autonoma. Per ulteriori informazioni su questo argomento, consulta Esegui un'attività autonoma nella documentazione di Amazon ECS.	Amministratore di sistema, sviluppatore

Verifica i CloudWatch registri

Operazione	Description	Competenze richieste
Verifica i registri.	1. Apri la CloudWatch console, scegli Gruppi di log, quindi scegli `/aws/ecs/containerinsights/{{cluster_ARN}}/firelens/application` .	Amministratore di sistema, sviluppatore

Operazione	Description	Competenze richieste
	2. Verifica i log, in particolare i campi personalizzati aggiunti dal parser personalizzato. 3. Utilizzato CloudWatch per filtrare i log in base ai campi personalizzati.	

Risorse correlate

- Nozioni di base su Docker per Amazon ECS
- Amazon ECS su AWS Fargate
- Configurazione dei parametri di base del servizio

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Crea un framework di orchestrazione delle risorse basato su API utilizzando Actions e Terragrunt GitHub

Tamilselvan P, Abhigyan Dandriyal, Sandeep Gawande e Akash Kumar, Amazon Web Services

Riepilogo

Questo modello sfrutta i flussi di lavoro di GitHub Actions per automatizzare l'approvvigionamento delle risorse tramite payload JSON standardizzati, eliminando la necessità di una configurazione manuale. Questa pipeline automatizzata gestisce l'intero ciclo di vita dell'implementazione e può integrarsi perfettamente con vari sistemi di frontend, dai componenti dell'interfaccia utente personalizzati a. ServiceNow La flessibilità della soluzione consente agli utenti di interagire con il sistema tramite le loro interfacce preferite, mantenendo processi standardizzati.

L'architettura configurabile della pipeline può essere adattata per soddisfare diversi requisiti organizzativi. L'implementazione di esempio si concentra sul provisioning di Amazon Virtual Private Cloud (Amazon VPC) e Amazon Simple Storage Service (Amazon S3). Il modello affronta efficacemente le sfide comuni di gestione delle risorse cloud standardizzando le richieste all'interno dell'organizzazione e fornendo punti di integrazione coerenti. Questo approccio semplifica la richiesta e la gestione delle risorse da parte dei team, garantendo al contempo la standardizzazione.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un GitHub account attivo con accesso al repository configurato

Limitazioni

- Le nuove risorse richiedono l'aggiunta manuale di `terragrunt.hcl` file alla configurazione del repository.
- Alcune Servizi AWS non sono disponibili in tutte Regioni AWS. Per la disponibilità regionale, vedi AWS Servizi per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Architecture

Il diagramma seguente mostra i componenti e il flusso di lavoro di questo modello.

Il diagramma dell'architettura mostra le seguenti azioni:

1. L'utente invia un payload JSON a GitHub Actions, attivando la pipeline di automazione.
2. La pipeline GitHub Actions recupera il codice delle risorse richiesto dai repository Terragrunt e Terraform, in base alle specifiche del payload.
3. La pipeline assume il ruolo (IAM) appropriato utilizzando l'ID specificato. AWS Identity and Access Management Account AWS Quindi, la pipeline distribuisce le risorse sulla destinazione Account AWS e gestisce lo stato di Terraform utilizzando il bucket Amazon S3 specifico dell'account e la tabella Amazon DynamoDB.

Ciascuno Account AWS contiene ruoli IAM per un accesso sicuro, un bucket Amazon S3 per lo storage dello stato Terraform e una tabella DynamoDB per il blocco dello stato. Questo design consente la distribuzione controllata e automatizzata delle risorse su tutto il territorio. Account AWS Il processo di distribuzione mantiene la corretta gestione dello stato e il controllo degli accessi tramite bucket Amazon S3 dedicati e ruoli IAM in ogni account.

Tools (Strumenti)

Servizi AWS

- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.

Altri strumenti

- GitHub Actions è una piattaforma di integrazione e distribuzione continua (CI/CD) strettamente integrata con i repository. GitHub Puoi utilizzare GitHub Actions per automatizzare la pipeline di compilazione, test e distribuzione.
- Terraform è uno strumento di infrastruttura come codice (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.
- Terragrunt è uno strumento di orchestrazione che estende sia le funzionalità di Terraform che quelle di Terraform. OpenTofu Gestisce il modo in cui vengono applicati i modelli di infrastruttura generici, semplificando la scalabilità e la manutenzione di grandi infrastrutture.

Archivio di codice

Il codice per questo modello è disponibile nel repository GitHub sample-aws-orchestration-pipeline-terraform.

Best practice

- Archivia AWS le credenziali e i dati sensibili utilizzando i segreti del GitHub repository per un accesso sicuro.
- Configura il provider OpenID Connect (OIDC) for GitHub Actions per assumere il ruolo IAM, evitando credenziali statiche.
- Segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.

Epiche

Crea e configura il repository

Operazione	Description	Competenze richieste
Inizializza il GitHub repository.	Per inizializzare il GitHub repository, utilizzare i seguenti passaggi: 1. Crea un nuovo GitHub repository per ospitare il codice della pipeline.	DevOps ingegnere

Operazione	Description	Competenze richieste
	2. Importa i `workflow-trigger.yml` file `deployment.yml` and che si trovano nella cartella. github/workflows del repository di origine.	
Configura i ruoli e le autorizzazioni IAM.	Per configurare i ruoli e le autorizzazioni IAM, utilizza i seguenti passaggi: 1. Crea un ruolo IAM con le relazioni di fiducia necessari e affinché GitHub Actions si connetta alle azioni AWS utilizzando un provider di identità (IdP) OpenID Connect (OIDC). 2. Associa le autorizzazioni richieste al ruolo IAM per creare il backend e le risorse desiderate. Per ulteriori informazioni, consulta un esempio di policy per creare un VPC con Amazon VPC insieme al bucket di backend Amazon S3 e alla tabella DynamoDB.	DevOps ingegnere

Operazione	Description	Competenze richieste

Imposta GitHub segreti e variabili.	Per istruzioni su come impostare i segreti e le variabili del repository nel GitHub repository, consulta Creazione di variabili di configurazione per un repository nella documentazione. GitHub Configura le seguenti variabili:
	• Segreti del repository
	  • `PAT_TOKEN` — Il tuo token di accesso personale con le autorizzazioni per eseguire operazioni GitHub
	• Variabili del repository
	  • `aws_role`— Il ruolo IAM Amazon Resource Name (ARN) con le autorizzazioni appropriate per distribuire le risorse richieste e collegare GitHub le azioni alle azioni in AWS
		DevOps ingegnere

Operazione	Description	Competenze richieste
Crea la struttura del repository.	Per creare la struttura del repository, attenersi alla seguente procedura: 1. Crea una nuova cartella nel `main` ramo per archiviare il `terragrunt.hcl` file, nonché gli output e i log delle modifiche dopo la creazione delle risorse. 2. Assegna un nome alla cartella in lettere minuscole in base al tipo di risorsa a cui desideri effettuare il provisioning tramite essa. La cartella viene utilizzata così com'è più avanti nel payload. Per ulteriori informazioni, consulta una struttura di esempio per Amazon S3 e Amazon VPC nel repository.	DevOps ingegnere

Attiva la pipeline e convalida i risultati

Operazione	Description	Competenze richieste
Esegui la pipeline usando curl.	Per eseguire la pipeline utilizzando curl, utilizzate i seguenti passaggi: 1. Crea un file payload.json nella tua directory locale. Segui la struttura definita	DevOps ingegnere

Operazione	Description	Competenze richieste
	per il file payload e stringila prima di inviare la richiesta . Per ulteriori informazioni, consulta i payload di esempio nel repository. 2. Inviate la richiesta sul vostro terminale utilizzando l' GitHub API, come illustrato nell'esempio seguente: `curl -X POST \ -H "Accept: application/vnd.github.v3+json" \ -H "Authorization: token YOUR_GITHUB_TOKEN" \ -d @payload.json \ https://api.github.com/repos/OWNER/REPO/actions/workflows/workflow-trigger.yml/dispatches` Nell'esempio, fornisci i tuoi valori per quanto segue: • `YOUR_GITHUB_TOKEN` con il tuo token di accesso GitHub personale • `OWNER`con il nome del proprietario del repository • `REPO`con il nome del repository	

Operazione	Description	Competenze richieste
	Per ulteriori informazioni sul processo di esecuzione della pipeline, vedere Informazioni aggiuntive.	

Operazione	Description	Competenze richieste
Convalida i risultati dell'esecuzione della pipeline	Per convalidare i risultati, utilizzate i seguenti passaggi: 1. Monitora l'esecuzione del flusso di lavoro GitHub Actions nella scheda Azioni del tuo repository. 2. Dopo la corretta esecuzione dei flussi di lavoro, verifica la creazione delle risorse utilizzando Console di gestione AWS quanto segue: a. Per Amazon VPC: • Accedi al servizio Amazon VPC nel campo specificato. Regione AWS • Verifica la presenza di un nuovo VPC con tag che corrispondano ai parametri della richiesta. • Verifica il blocco CIDR e altre configurazioni. b. Per Amazon S3: • Passa ai bucket Amazon S3. • Nei bucket generici, verifica la presenza di un nuovo bucket che corrisponda	DevOps ingegnere

Operazione	Description	Competenze richieste
	ai parametri della richiesta. • Verifica il nome del bucket S3 e altre configurazioni. Puoi anche verificare in modo incrociato le risorse create utilizzando il `output.json` file creato nel repository che si trova all'interno della stessa risorsa del file. `terragrun t.hcl`	

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Invia una richiesta di pulizia.	Per eliminare le risorse che non sono più necessarie, procedi nel seguente modo: 1. Inviate la richiesta di eliminazione utilizzando lo stesso endpoint API, ma modificate il payload come segue: • Passa `RequestType a delete` in. `RequestPa rameters` • Mantieni tutti gli altri parametri identici alla `create` richiesta.	DevOps ingegnere

Operazione	Description	Competenze richieste
	2. Monitora il processo di eliminazione in GitHub Azioni. 3. Dopo il completamento del flusso di lavoro, verifica che il `changelog.json` file all'interno della cartella delle risorse mostri lo stato di`deleted`. 4. Verifica la rimozione delle risorse utilizzando Console di gestione AWS.	

Risorse correlate

AWS Blog

- Usa i ruoli IAM per connettere GitHub le azioni alle azioni in AWS

Servizio AWS documentazione

- creazione di ruoli IAM
- Monitoraggio dei file di CloudTrail registro con Amazon CloudWatch Logs
- Best practice di sicurezza per Amazon S3

GitHub resources

- Crea un evento di invio del repository
- Creazione di webhook
- Implementa forti controlli di accesso sul repository GitHub
- Verifica regolarmente l'accesso al repository
- Controlli di sicurezza in corso CI/CD
- Utilizza l'autenticazione a più fattori per gli account GitHub

Informazioni aggiuntive

Processo di esecuzione della pipeline

Di seguito sono riportati i passaggi dell'esecuzione della pipeline:

1. Convalida il formato del payload JSON: assicura che la configurazione JSON in entrata sia strutturata correttamente e contenga tutti i parametri richiesti
2. Presuppone il ruolo IAM specificato: autentica e assume il ruolo IAM richiesto per le operazioni AWS
3. Scarica il codice Terraform e Terragrunt richiesto: recupera la versione specificata del codice risorsa e delle dipendenze
4. Esegue la distribuzione delle risorse: applica la configurazione per distribuire o aggiornare le risorse nell'ambiente di destinazione AWS

Payload di esempio utilizzato per la creazione di VPC

Di seguito è riportato un esempio di codice per la creazione di bucket di stato del backend Terraform:

```
state_bucket_name = "${local.payload.ApplicationName}-${local.payload.EnvironmentId}-
tfstate"
```

```
lock_table_name = "${local.payload.ApplicationName}-${local.payload.EnvironmentId}-
tfstate-lock"
```

Di seguito è riportato un esempio di payload per la creazione di un VPC con Amazon VPC, `vpc_cidr` in cui definisce le specifiche del blocco CIDR per il VPC. Il bucket di stato Terraform è mappato su una variabile definita nei file. `terraform` Il `ref` parametro contiene il nome del ramo del codice da eseguire.

```
{
    "ref": "main",
    "inputs": {
        "RequestParameters": {
            "RequestId": "1111111",
            "RequestType": "create",
            "ResourceType": "vpc",
            "AccountId": "1234567890",
            "AccountAlias": "account-alias",
```

```
            "RegionId": "us-west-2",
            "ApplicationName": "myapp",
            "DivisionName": "division-name",
            "EnvironmentId": "dev",
            "Suffix": "poc"
        },
        "ResourceParameters": [
            {
                "VPC": {
                    "vpc_cidr": "10.0.0.0/16"
                }
            }
        ]
    }
}
```

`RequestParameters`vengono utilizzati per tenere traccia dello stato della richiesta nella sezione pipeline e `tfstate` vengono creati sulla base di queste informazioni. I seguenti parametri contengono metadati e informazioni di controllo:

- `RequestId`— Identificatore univoco per la richiesta
- `RequestType`— Tipo di operazione (creazione, aggiornamento o eliminazione)
- `ResourceType`— Tipo di risorsa da fornire
- `AccountId`— Obiettivo Account AWS di implementazione
- `AccountAlias`— Nome descrittivo per Account AWS
- `RegionId`— Regione AWS per la distribuzione delle risorse
- `ApplicationName`— Nome dell'applicazione
- `DivisionName`— Divisione organizzativa
- `EnvironmentId`— Ambiente (ad esempio, dev e prod)
- `Suffix`— Identificatore aggiuntivo per le risorse

`ResourceParameters`contengono una configurazione specifica della risorsa che si associa alle variabili definite nei file Terraform. Tutte le variabili personalizzate che devono essere passate ai moduli Terraform devono essere incluse in. `ResourceParameters` Il parametro `vpc_cidr` è obbligatorio per Amazon VPC.

Crea richieste pull automatizzate per l' AWS infrastruttura gestita da Terraform utilizzando Actions GitHub

Matt Padgett, Ashish Bhatt, Ashwin Divakaran, Sandip Gangapadhyay e Prafful Gupta, Amazon Web Services

Riepilogo

Questo modello presenta un'utilità di automazione progettata per eliminare il lavoro manuale e ripetitivo necessario nella gestione delle modifiche su più repository Terraform. Molte organizzazioni utilizzano i repository Terraform per gestire la propria infrastruttura come codice (IaC), spesso con centinaia di repository separati che rappresentano ambienti, servizi o team diversi. La gestione di questi repository su larga scala rappresenta una sfida operativa significativa. Le attività di routine come l'aggiornamento di un parametro, l'aggiornamento delle versioni dei moduli o l'applicazione di modifiche alla configurazione spesso richiedono la creazione e la gestione delle richieste pull (PRs) su molti repository più volte al giorno.

Anche per modifiche semplici, questo processo ripetitivo e manuale richiede molto tempo ed è soggetto a errori. Gli ingegneri devono applicare costantemente la stessa modifica a tutti gli archivi mirati e creare titoli e descrizioni PR significativi. Inoltre, spesso devono interagire con strumenti esterni come Jira per recuperare o includere riferimenti per il monitoraggio dei problemi. Queste attività, sebbene necessarie, sono un sollevamento indifferenziato di carichi pesanti che richiedono tempo prezioso di progettazione e riducono l'efficienza complessiva. La mancanza di automazione in questo flusso di lavoro crea attriti, rallenta l'erogazione e aumenta il carico cognitivo sui team incaricati di mantenere infrastrutture Terraform su larga scala.

Panoramica della soluzione

Per affrontare questa sfida, questo modello offre un'utilità interamente basata sulla configurazione, che consente agli utenti di definire le modifiche desiderate in un file di configurazione strutturato. Questo file specifica i repository, i moduli, i parametri e i valori di destinazione utilizzando uno schema chiaramente definito.

Una volta configurata, l'utilità esegue i seguenti passaggi automatici:

1. Legge la configurazione definita dall'utente per determinare l'ambito e la natura delle modifiche
2. Crea un nuovo ramo in ogni repository di destinazione con gli aggiornamenti richiesti applicati
3. Genera un PR per ogni modifica, garantendo la coerenza tra tutti gli archivi

4. Invia notifiche Slack (opzionale) per avvisare le parti interessate con link diretti al file creato PRs

Automatizzando queste attività ripetitive, l'utilità riduce in modo significativo il tempo, l'impegno e i rischi associati alla gestione degli aggiornamenti dell'infrastruttura su larga scala. Consente ai team di concentrarsi su attività ingegneristiche di maggior valore, garantendo al contempo che le modifiche vengano applicate in modo coerente e possano essere tracciate in tutti gli archivi.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Python versione 3.8 o successiva.
- Un token di accesso GitHub personale (PAT). Per ulteriori informazioni, consulta Creazione di un token di accesso personale (classico) nella GitHub documentazione.
- Il GitHub PAT può accedere ai repository di destinazione in modo che l'utilità possa eseguire operazioni come la creazione di rami e richieste pull. Per ulteriori informazioni, consultate l'archivio del GitHub codice di questo pattern.

Limitazioni

- La complessità della configurazione rappresenta la sfida principale. L'efficacia dell'automazione è limitata dalle funzionalità del relativo file di configurazione. Sebbene il sistema gestisca le modifiche standard in modo efficiente, le modifiche complesse dell'infrastruttura potrebbero richiedere un intervento manuale e alcuni casi limite non rientrano nell'ambito della gestione automatizzata.
- La sicurezza e l'accesso presentano considerazioni importanti, in particolare nella gestione dei token di GitHub accesso e dei limiti di velocità delle API. Le organizzazioni devono bilanciare attentamente la necessità di automazione con l'archiviazione e la gestione sicure delle credenziali, garantendo controlli di accesso adeguati e mantenendo l'efficienza operativa.
- I vincoli di convalida rappresentano un'altra limitazione importante perché il sistema automatizzato ha una capacità limitata di convalidare la logica aziendale e i requisiti specifici dell'ambiente. Le dipendenze complesse e le interazioni tra servizi spesso richiedono la supervisione umana, poiché la convalida automatizzata non è in grado di cogliere appieno tutte le sfumature contestuali e le regole aziendali.
- Quando si affrontano modifiche all'infrastruttura su larga scala emergono problemi di scalabilità e prestazioni. Il sistema deve funzionare entro i limiti delle GitHub API e gestire

contemporaneamente numerosi repository. Le operazioni che richiedono un uso intensivo di risorse su un'infrastruttura estesa possono creare colli di bottiglia prestazionali che richiedono una gestione attenta.

- I limiti di integrazione limitano la flessibilità del sistema perché è progettato principalmente per funzionare con strumenti specifici come e Slack. GitHub Le organizzazioni che utilizzano strumenti diversi potrebbero aver bisogno di soluzioni personalizzate e le opzioni di personalizzazione del flusso di lavoro di questo modello sono limitate ai punti di integrazione supportati.

Architecture

Il diagramma seguente mostra il flusso di lavoro e i componenti di questa soluzione.

Il flusso di lavoro consiste nei seguenti passaggi:

1. Lo sviluppatore attiva GitHub Actions specificando il repository Terraform.
2. L'utilità di automazione legge le configurazioni definite.
3. L'utilità di automazione richiama anche il repository Terraform fornito.
4. L'utilità di automazione crea un nuovo ramo e aggiorna i modelli Terraform localmente.
5. L'utilità di automazione invia il nuovo ramo al repository e crea un nuovo PR.
6. L'utilità di automazione utilizza le notifiche Slack che includono link PR per avvisare gli sviluppatori e abilita i modelli Terraform per la distribuzione. Cloud AWS

Tools (Strumenti)

- GitHubè una piattaforma per sviluppatori che gli sviluppatori possono utilizzare per creare, archiviare, gestire e condividere il proprio codice.
- GitHub Actions è una piattaforma di integrazione e distribuzione continua (CI/CD) strettamente integrata con i repository. GitHub Puoi utilizzare GitHub Actions per automatizzare la pipeline di compilazione, test e distribuzione.
- HashiCorp Terraform è uno strumento di infrastruttura come codice (IaC) che ti aiuta a creare e gestire risorse cloud e locali.
- Slack, un'offerta di Salesforce, è una piattaforma conversazionale basata sull'intelligenza artificiale che fornisce collaborazione tramite chat e video, automatizza i processi senza codice e supporta la condivisione delle informazioni.

Archivio di codice

Il codice per questo modello è disponibile nell'archivio GitHub Automated Terraform Infrastructure Update Workflow using GitHub Actions.

Best practice

- Una gestione efficace delle modifiche è fondamentale per un'implementazione di successo. Le organizzazioni dovrebbero adottare una strategia di implementazione graduale per cambiamenti su larga scala. Mantieni convenzioni di denominazione delle filiali e descrizioni delle pubbliche relazioni coerenti e assicurati una documentazione completa di tutte le modifiche.
- I controlli di sicurezza devono essere implementati in modo rigoroso, concentrandosi sui principi di accesso con privilegi minimi e sulla gestione sicura delle credenziali. Abilita le regole di protezione delle filiali per prevenire modifiche non autorizzate. Effettua controlli di sicurezza regolari per mantenere l'integrità del sistema.
- Un protocollo di test robusto dovrebbe includere `terraform plan` l'esecuzione automatizzata in pipeline di integrazione continua e distribuzione continua (CI/CD). Il protocollo dovrebbe includere anche controlli di convalida prima del commit e ambienti di revisione dedicati per le modifiche critiche. Questo approccio di test a più livelli aiuta a catturare tempestivamente i problemi e garantisce la stabilità dell'infrastruttura.
- La strategia di monitoraggio deve comprendere meccanismi di allarme completi, tracciamento dettagliato delle success/failure metriche e meccanismi di riprova automatizzati in caso di operazioni non riuscite. Questa strategia aiuta a garantire la visibilità operativa e consente una risposta rapida a qualsiasi problema che si presenti.
- Gli standard di configurazione dovrebbero enfatizzare il controllo delle versioni per tutte le configurazioni, mantenendo la modularità per la riusabilità e la scalabilità. Una documentazione chiara dello schema e degli esempi aiuta i team a comprendere e utilizzare il sistema di automazione in modo efficace.

Epiche

Installazione e configurazione

Operazione	Description	Competenze richieste
Configura il repository.	Per configurare il repository, esegui i seguenti comandi:	AWS DevOps

Operazione	Description	Competenze richieste

	`# Clone the automation tool repository`
	`git clone https://github.com/aws-samples/sample-terraform-pr-automation-utility`
	`cd sample-terraform-pr-automation-utility`
	
	`# Copy example configuration`
	`cp config.example.yaml config.yaml`	

Installare le dipendenze.	Per installare e verificare le dipendenze di Python, esegui i seguenti comandi:
	
	`# Install Python dependencies`
	`pip3 install -r requirements.txt`
	
	`# Verify installation`
	`python3 -c "import github; import hcl2; import yaml; import requests; print('All packages installed successfully')"`
		AWS DevOps

Operazione	Description	Competenze richieste

Configura il GitHub token.	Per configurare il GitHub token e quindi verificarne il funzionamento, esegui i seguenti comandi:
	`# Set GitHub token environment variable`
	`export GITHUB_TOKEN="your_github_token_here"`
	
	`# Verify token works`
	`curl -H "Authorization: token $GITHUB_TOKEN" https://api.github.com/user`
		AWS DevOps

Imposta il file di configurazione per le modifiche a Terraform

Operazione	Description	Competenze richieste

Configura il `config.yaml` file.	Per definire i repository di destinazione e le modifiche desiderate, modificate il file `config.yaml` come segue:
	`repositories:`
	`  - owner: "your-org"`
	`  repo: "your-terraform-repo"`
	`  files:`
	`  - path: "variables.tf"`
	`  changes:`
	`  variables:`
	`  - app_version:`
		AWS DevOps

Operazione	Description	Competenze richieste
	default: update: - from: ["1.0.0"] to: "1.1.0" settings: pr_title_template: "Infrastructure Update - {{timestamp}}" slack: username: "Terrafor m Bot" icon_emoji: ":terraform:" notify_on_success: true notify_on_error: true notify_batch_summa ry: true	

Testa e convalida

Operazione	Description	Competenze richieste
Effettua i test prima del volo.	Verifica sempre la tua configurazione prima di eseguirla nei repository di produzione. Utilizza il seguente comando: # 1. Test configuration syntax python3 -c "from main import get_confi	AWS DevOps

Operazione	Description	Competenze richieste

	`g_content; get_confi g_content()"`
	
	`# 2. Run in dry-run mode first`
	`DRY_RUN=true python3 main.py`
	
	`# 3. Test with minimal configuration`
	`# Use a simple config.ya ml with just one repository and one change`	

Verifica l'accesso al repository.	Per verificare che il GitHub token possa accedere al repository, esegui il seguente comando:
	
	`# Test GitHub token access`
	`curl -H "Authoriz ation: token $GITHUB_T OKEN" \`
	`https://api.github .com/repos/owner/r epo-name`
	
	`# Should return repository informati on, not 404`
		AWS DevOps

Esegui l'utilità di automazione

Operazione	Description	Competenze richieste
Esegui l'utilità di automazione utilizzando l'interfaccia utente GitHub Actions.	Per eseguire l'utilità di automazione utilizzando l'interfaccia utente GitHub Actions, procedi come segue: 1. Accedi al tuo repository su. GitHub 2. Scegli la scheda Azioni. 3. Scegli il flusso di lavoro Terraform Infrastructure Update Automation. 4. Scegli Esegui flusso di lavoro. 5. Configura i seguenti input del flusso di lavoro: • Configurazione del codice sorgente: • Seleziona il ramo di destinazione per l'automazione. • Specificate il percorso del file di configura zione (`config.ya ml `) che contiene le modifiche desiderate. • Controlli di anteprima: • Scegliete l'opzione Anteprima per rivedere le modifiche senza applicarle. • Gestione delle filiali:	AWS DevOps

Operazione	Description	Competenze richieste
	• Inserisci il ramo base per creare nuovi rami di funzionalità. • Immettete la configura zione del prefisso del ramo. • Seleziona la casella di controllo Chiudi automaticamente le richieste pull obsolete. • Configurazione delle notifiche: • Inserisci l'URL per l'URL del webhook di Slack per le notifiche complete (sostituisce il segreto del repository). • Scegli la casella di controllo Verifica l'integrazione con Slack prima dell'elaborazione (solo esecuzione a secco). • Inserisci le tue informazioni per un canale Slack predefini to di Override (ad esempio, #infrastr ucture -test). • Impostazioni avanzate: • Scegli la casella di controllo Abilita la registrazione di debug	

Operazione	Description	Competenze richieste
	per la risoluzione dei problemi.	

(Alternativa) Esegui l'utilità di automazione dalla riga di comando.	Se preferisci, puoi eseguire l'utilità di automazione dalla riga di comando anziché utilizzando l'interfaccia utente GitHub Actions. Utilizza il seguente comando:
	`# Run actual automation`
	`python3 main.py`
		AWS DevOps

Convalida PRs e modifiche

Operazione	Description	Competenze richieste
Rivedi le creazioni PRs e le modifiche.	Per monitorare i risultati dell'esecuzione del GitHub flusso di lavoro, procedi come segue: • Controllate lo stato di elaborazione dei registri di esecuzione del flusso di lavoro. • Controllate PRs il file creato dall'utilità di automazione. • Monitora le notifiche di Slack (se configurate).	AWS DevOps

Eliminazione

Operazione	Description	Competenze richieste
(Facoltativo) Pulisci PRs.	Chiusura abbandonata o non necessaria PRs.	AWS DevOps

Risorse correlate

AWS Guida prescrittiva

- Utilizzo di Terraform come strumento IaC per Cloud AWS

GitHub documentazione

- Informazioni sulle pull request
- Gestione dei token di accesso personali
- Comprensione delle GitHub azioni
- Guida rapida alle azioni GitHub

Crea automaticamente pipeline CI dinamiche per progetti Java e Python

Aromal Raj Jayarajan, Vijesh Vijayakumaran Nair, MAHESH RAGHUNANDANAN e Amarnath Reddy, Amazon Web Services

Riepilogo

Questo modello mostra come creare automaticamente pipeline dinamiche di integrazione continua (CI) per progetti Java e Python utilizzando gli strumenti di sviluppo AWS.

Con la diversificazione degli stack tecnologici e l'aumento delle attività di sviluppo, può diventare difficile creare e mantenere pipeline CI coerenti all'interno di un'organizzazione. Automatizzando il processo in AWS Step Functions, puoi assicurarti che le tue pipeline CI siano coerenti nel loro utilizzo e approccio.

Per automatizzare la creazione di pipeline CI dinamiche, questo modello utilizza i seguenti input variabili:

- Linguaggio di programmazione (solo Java o Python)
- Nome della pipeline
- Fasi della pipeline richieste

> **Note**
>
> Step Functions orchestra la creazione di pipeline utilizzando più servizi AWS. Per ulteriori informazioni sui servizi AWS utilizzati in questa soluzione, consulta la sezione Strumenti di questo modello.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un bucket Amazon S3 nella stessa regione AWS in cui viene distribuita questa soluzione
- Un principal AWS Identity and Access Management (IAM) con CloudFormation le autorizzazioni AWS necessarie per creare le risorse necessarie per questa soluzione

Limitazioni

- Questo modello supporta solo progetti Java e Python.
- I ruoli IAM forniti in questo modello seguono il principio del privilegio minimo. Le autorizzazioni dei ruoli IAM devono essere aggiornate in base alle risorse specifiche che la pipeline CI deve creare.

Architecture

Stack tecnologico Target

- AWS CloudFormation
- AWS CodeBuild
- AWS CodeCommit
- AWS CodePipeline
- IAM
- Amazon Simple Storage Service (Amazon S3)
- AWS Systems Manager
- AWS Step Functions
- AWS Lambda
- Amazon DynamoDB

Architettura Target

Il diagramma seguente mostra un esempio di flusso di lavoro per la creazione automatica di pipeline CI dinamiche per progetti Java e Python utilizzando gli strumenti di sviluppo AWS.

Il diagramma mostra il flusso di lavoro seguente:

1. Un utente AWS fornisce i parametri di input per la creazione di pipeline CI in formato JSON. Questo input avvia un flusso di lavoro Step Functions (macchina a stati) che crea una pipeline CI utilizzando gli strumenti di sviluppo AWS.
2. Una funzione Lambda legge una cartella denominata input-reference, archiviata in un bucket Amazon S3, e quindi genera un file buildspec.yml. Questo file generato definisce le fasi della pipeline CI e viene archiviato nello stesso bucket Amazon S3 che memorizza i riferimenti ai parametri.

3. Step Functions controlla le dipendenze del flusso di lavoro di creazione della pipeline CI per eventuali modifiche e aggiorna lo stack di dipendenze secondo necessità.
4. Step Functions crea le risorse della pipeline CI in uno CloudFormation stack, tra cui un CodeCommit repository, un CodeBuild progetto e una pipeline. CodePipeline
5. Lo CloudFormation stack copia il codice sorgente di esempio per lo stack tecnologico selezionato (Java o Python) e il file buildspec.yml nel repository. CodeCommit
6. I dettagli del runtime della pipeline CI sono archiviati in una tabella DynamoDB.

Automazione e scalabilità

- Questo modello è destinato all'uso in un solo ambiente di sviluppo. Le modifiche alla configurazione sono necessarie per l'uso in più ambienti di sviluppo.
- Per aggiungere il supporto per più di uno CloudFormation stack, puoi creare CloudFormation modelli aggiuntivi. Per ulteriori informazioni, consulta Getting started with AWS CloudFormation nella CloudFormation documentazione.

Tools (Strumenti)

Strumenti

- AWS Step Functions è un servizio di orchestrazione serverless che ti aiuta a combinare le funzioni di AWS Lambda e altri servizi AWS per creare applicazioni aziendali critiche.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- AWS CodeBuild è un servizio di build completamente gestito che ti aiuta a compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS CodeCommit è un servizio di controllo delle versioni che consente di archiviare e gestire archivi Git in modo privato, senza dover gestire il proprio sistema di controllo del codice sorgente.
- AWS ti CodePipeline aiuta a modellare e configurare rapidamente le diverse fasi di un rilascio di software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.

- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account e regioni AWS.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- AWS Systems Manager Parameter Store fornisce uno storage sicuro e gerarchico per la gestione dei dati di configurazione e la gestione dei segreti.

Codice

Il codice per questo pattern è disponibile nel repository. GitHub automated-ci-pipeline-creation Il repository contiene i CloudFormation modelli necessari per creare l'architettura di destinazione delineata in questo modello.

Best practice

- Non inserire credenziali (segrete) come token o password direttamente nei CloudFormation modelli o nelle configurazioni di azione Step Functions. In tal caso, le informazioni verranno visualizzate nei log di DynamoDB. Utilizza invece AWS Secrets Manager per configurare e archiviare segreti. Quindi, fai riferimento ai segreti archiviati in Secrets Manager all'interno dei CloudFormation modelli e delle configurazioni di azione di Step Functions, se necessario. Per ulteriori informazioni, consulta Cos'è AWS Secrets Manager nella documentazione di Secrets Manager.
- Configura la crittografia lato server per gli CodePipeline artefatti archiviati in Amazon S3. Per ulteriori informazioni, consulta Configurare la crittografia lato server per gli artefatti archiviati in Amazon S3 nella documentazione. CodePipeline CodePipeline
- Applica le autorizzazioni con privilegi minimi durante la configurazione dei ruoli IAM. Per ulteriori informazioni, consulta Applicazione delle autorizzazioni del privilegio minimo nella documentazione di IAM.
- Assicurati che il tuo bucket Amazon S3 non sia accessibile al pubblico. Per ulteriori informazioni, consulta Configurazione dell'impostazione di accesso pubblico a blocchi per i bucket S3 nella documentazione di Amazon S3.

- Assicurati di attivare il controllo delle versioni per il tuo bucket Amazon S3. Per ulteriori informazioni, consulta Usare il controllo delle versioni nei bucket S3 nella documentazione di Amazon S3.
- Usa IAM Access Analyzer per configurare le policy IAM. Lo strumento fornisce consigli pratici per aiutarti a creare policy IAM sicure e funzionali. Per ulteriori informazioni, consulta Using AWS Identity and Access Management Access Analyzer nella documentazione IAM.
- Quando possibile, definisci condizioni di accesso specifiche durante la configurazione delle policy IAM.
- Attiva la CloudWatch registrazione di Amazon per scopi di monitoraggio e controllo. Per ulteriori informazioni, consulta What is Amazon CloudWatch Logs? nella CloudWatch documentazione.

Epiche

Configura i prerequisiti

Operazione	Description	Competenze richieste
Creare un bucket Amazon S3.	Crea un bucket Amazon S3 (o usa un bucket esistente) per archiviare i CloudFormation modelli, il codice sorgente e i file di input richiesti per la soluzione. Per ulteriori informazioni, consulta Fase 1: Crea il tuo primo bucket S3 nella documentazione di Amazon S3. **Note** Il bucket Amazon S3 deve trovarsi nella stessa regione AWS in	AWS DevOps

Operazione	Description	Competenze richieste
	cui stai distribuendo la soluzione.	
Clona il GitHub repository.	Clona il GitHub automated-ci-pipeline-creationrepository eseguendo il seguente comando in una finestra di terminale: `git clone https://github.com/aws-samples/automated-ci-pipeline-creation.git` Per ulteriori informazioni, consulta Clonazione di un repository nella documentazione. GitHub	AWS DevOps

Operazione	Description	Competenze richieste
Carica la cartella Solution Templates dal GitHub repository clonato nel tuo bucket Amazon S3.	Copia i contenuti dalla cartella Solution-Templates clonata e caricali nel bucket Amazon S3 che hai creato. Per ulteriori informazioni, consulta Caricamento di oggetti nella documentazione di Amazon S3. **Note** Assicurati di caricare solo il contenuto della cartella Solution-Templates. Puoi caricare i file solo a livello root del bucket Amazon S3.	AWS DevOps

Implementazione della soluzione

Operazione	Description	Competenze richieste
Crea uno CloudFormation stack per distribuire la soluzione utilizzando il file template.yml nel repository clonato. GitHub	1. Accedi alla Console di gestione AWS e apri la CloudFormation console AWS. 2. Seleziona Crea stack. Viene visualizzato un elenco a discesa. 3. Nell'elenco a discesa, seleziona Con nuove risorse (standard). Viene	Amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste
	visualizzata la pagina Crea stack. 4. Nella sezione Specifica re il modello, seleziona la casella di controllo accanto a Carica un file modello. 5. Selezionare Choose file (Scegli file). Quindi, accedi alla cartella principale del GitHub repository clonato e seleziona il file template. yml. Scegliere quindi Open (Apri). 6. Scegli Next (Successivo). Viene visualizzata la pagina Specificare i dettagli dello stack. 7. Nella sezione Parametri , specificare i seguenti parametri: • Per S3 TemplateB ucketName, inserisci il nome del bucket Amazon S3 creato in precedenza, che contiene il codice sorgente e i riferimenti per questa soluzione. Assicurati che il parametro del nome del bucket sia in minuscolo. • Per Dynamo DBTable, inserisci un nome per la tabella DynamoDB creata	

Operazione	Description	Competenze richieste
	dallo stack. CloudFormation • Per StateMachineName, inserisci un nome per la macchina a stati Step Functions creata dallo CloudFormation stack. 8. Scegli Next (Successivo). Viene visualizzata la pagina Configura le opzioni dello stack. 9. Nella pagina Configure stack options (Configura opzioni pila), scegliere Next (Successivo). Non modificate nessuno dei valori predefiniti. Viene visualizzata la pagina Revisione. 10. Rivedi le impostazioni di creazione dello stack. Quindi, scegli Crea stack per avviare lo stack. **Note** Durante la creazione, lo stack viene elencato nella pagina Stacks con lo stato CREATE_IN_PROGRESS. Assicurati di attendere che lo stato dello stack	

Operazione	Description	Competenze richieste
	passi a CREATE_COMPLETE prima di completare i passaggi rimanenti di questo schema.	

Eseguire il test della configurazione

Operazione	Description	Competenze richieste
Esegui la funzione step che hai creato.	1. Accedi alla Console di gestione AWS e quindi apri la console Step Functions. 2. Apri la funzione Step che hai creato. 3. Selezionare Start execution (Avvia esecuzione). Quindi, inserisci i valori di input per il flusso di lavoro in formato JSON (vedi gli input di esempio seguenti). 4. Selezionare Start execution (Avvia esecuzione). Formattazione JSON `{ "details": { "tech_stack": "Name of the Tech Stack (python/java)", "project_name": "Name of the Project`	Amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste

```
 that you want to create
 with",
    "pre_build":
 "Choose the step if
 it required in the
 buildspec.yml file
 i.e., yes/no",
    "build": "Choose the
 step if it required in
 the buildspec.yml file
 i.e., yes/no",
    "post_build":
 "Choose the step if
 it required in the
 buildspec.yml file
 i.e., yes/no",
    "reports": "Choose
 the step if it required
 in the buildspec.yml
 file i.e., yes/no",
  }
}
```

Esempio di input Java JSON

```
{
  "details": {
    "tech_stack":
 "java",
    "project_name":
 "pipeline-java-pjt",
    "pre_build": "yes",
    "build": "yes",
    "post_build":
 "yes",
    "reports": "yes"
  }
}
```

Operazione	Description	Competenze richieste
	Esempio di input JSON in Python `{ "details": { "tech_stack": "python", "project_name": "pipeline-python-pjt", "pre_build": "yes", "build": "yes", "post_build": "yes", "reports": "yes" } }`	
Conferma che il CodeCommit repository per la pipeline CI è stato creato.	1. Accedi alla Console di gestione AWS e apri la CodeCommit console. 2. Nella pagina Repositories, verifica che il nome del CodeCommit repository che hai creato compaia nell'elenco dei repository. Al nome del repository viene aggiunto quanto segue: -Repo pipeline-java-pjt 3. Apri il CodeCommit repository e verifica che il codice sorgente di esempio insieme ai file buildspec.yml vengano inviati al ramo principale.	AWS DevOps

Operazione	Description	Competenze richieste
Controlla le risorse CodeBuild del progetto.	1. Accedi alla Console di gestione AWS e apri la CodeBuild console. 2. Nella pagina Crea progetti, verifica che il nome del CodeBuild progetto che hai creato compaia nell'elen co dei progetti. Al nome del progetto viene aggiunto quanto segue: pipeline-java-pjt -Build 3. Seleziona il nome del tuo CodeBuild progetto per aprirlo. Quindi, rivedi e convalida le seguenti configurazioni: • Configurazione del progetto • Origine • Ambiente • Specifiche di costruzione • Configurazione Batch • Artefatti	AWS DevOps

Operazione	Description	Competenze richieste
Convalida le CodePipeline fasi.	1. Accedi alla Console di gestione AWS e apri la CodePipeline console. 2. Nella pagina Pipelines, verifica che il nome della pipeline che hai creato compaia nell'elenco delle pipeline. Al nome della pipeline viene aggiunto quanto segue: -Pipeline pipeline-java-pjt 3. Seleziona il nome della pipeline per aprirla. Quindi, esamina e convalida ogni fase della pipeline, inclusi Commit e Deploy.	AWS DevOps
Verifica che la pipeline CI sia stata eseguita correttamente.	1. Nella CodePipeline console, nella pagina Pipelines, seleziona il nome della pipeline per visualizzarne lo stato. 2. Verifica che ogni fase della pipeline abbia lo stato Operato con successo.	AWS DevOps

Pulizia delle risorse

Operazione	Description	Competenze richieste
Elimina la pila di risorse inclusa. CloudFormation	Elimina lo stack di risorse della pipeline CI. CloudFormation	AWS DevOps

Operazione	Description	Competenze richieste
	Per ulteriori informazioni, consulta Eliminazione di uno stack sulla CloudFormation console AWS nella CloudFormation documentazione. **Note** Assicurati di eliminare lo stack denominato -stack. <project_name>	

Operazione	Description	Competenze richieste
Elimina le dipendenze della pipeline CI in Amazon S3 e. CloudFormation	1. Svuota il bucket Amazon S3 denominato. DeploymentArtifactBucket Per ulteriori informazioni, consulta Svuotare un bucket nella documentazione di Amazon S3. 2. Elimina lo stack di dipendenze della pipeline CI in. CloudFormation Per ulteriori informazioni, consulta Eliminazione di uno stack sulla CloudFormation console AWS nella CloudFormation documentazione. **Note** Assicurati di eliminare lo stack denominato. pipeline-creation-dependencies-stack	AWS DevOps

Operazione	Description	Competenze richieste
Elimina il bucket di modelli Amazon S3.	Elimina il bucket Amazon s3 che hai creato nella sezione Configura i prerequisiti di questo modello, che memorizza i modelli per questa soluzione. Per ulteriori informazioni, consulta Eliminazione di un bucket nella documentazione di Amazon S3.	AWS DevOps

Risorse correlate

- Creazione di una macchina a stati Step Functions che utilizza Lambda (documentazione AWS Step Functions)
- AWS Step Functions WorkFlow Studio (documentazione di AWS Step Functions)
- DevOps e AWS
- Come CloudFormation funziona AWS? (CloudFormation documentazione AWS)
- Completo CI/CD di AWS CodeCommit, AWS CodeBuild CodeDeploy, AWS e AWS CodePipeline (post sul blog AWS)
- Quote, requisiti di nome e limiti di caratteri IAM e AWS STS (documentazione IAM)

Implementa i canarini CloudWatch Synthetics utilizzando Terraform

Dhrubajyoti Mukherjee e Jean-François Landreau, Amazon Web Services

Riepilogo

È importante convalidare lo stato di un sistema dal punto di vista del cliente e confermare che i clienti siano in grado di connettersi. Ciò è più difficile quando i clienti non chiamano costantemente l'endpoint. Amazon CloudWatch Synthetics supporta la creazione di canaries, in grado di testare endpoint pubblici e privati. Utilizzando canaries, puoi conoscere lo stato di un sistema anche se non è in uso. Questi canarini sono script Node.js Puppeteer o script Python Selenium.

Questo modello descrive come utilizzare HashiCorp Terraform per distribuire canary che testano endpoint privati. Incorpora uno script Puppeteer che verifica se viene restituito un URL. `200-OK` Lo script Terraform può quindi essere integrato con lo script che distribuisce l'endpoint privato. Puoi anche modificare la soluzione per monitorare gli endpoint pubblici.

Prerequisiti e limitazioni

Prerequisiti

- Un account Amazon Web Services (AWS) attivo con un cloud privato virtuale (VPC) e sottoreti private
- L'URL dell'endpoint raggiungibile dalle sottoreti private
- Terraform installato nell'ambiente di distribuzione

Limitazioni

La soluzione attuale funziona per le seguenti versioni di runtime di CloudWatch Synthetics:

- syn-nodejs-puppeteer-3.4
- syn-nodejs-puppeteer-3,5
- syn-nodejs-puppeteer-3,6
- syn-nodejs-puppeteer-3,7

Man mano che vengono rilasciate nuove versioni di runtime, potrebbe essere necessario aggiornare la soluzione corrente. Sarà inoltre necessario modificare la soluzione per stare al passo con gli aggiornamenti di sicurezza.

Versioni del prodotto

- Terraform 1.3.0

Architecture

Amazon CloudWatch Synthetics è basato su CloudWatch, Lambda e Amazon Simple Storage Service (Amazon S3). Amazon CloudWatch offre una procedura guidata per creare i canarini e una dashboard che mostra lo stato delle corse dei canarini. La funzione Lambda esegue lo script. Amazon S3 archivia i log e gli screenshot delle corse Canary.

Questo modello simula un endpoint privato tramite un'istanza Amazon Elastic Compute Cloud (Amazon EC2) distribuita nelle sottoreti mirate. La funzione Lambda richiede interfacce di rete elastiche nel VPC in cui viene distribuito l'endpoint privato.

Il diagramma mostra:

1. Il canarino Synthetics avvia la funzione Lambda canary.
2. La funzione Lambda canary si connette all'interfaccia elastic network.
3. La funzione Lambda canary monitora lo stato dell'endpoint.
4. Il Synthetics Canary invia i dati di esecuzione al bucket e alle metriche S3. CloudWatch
5. Viene avviato un CloudWatch allarme in base alle metriche.
6. L' CloudWatch allarme avvia l'argomento Amazon Simple Notification Service (Amazon SNS).

Tools (Strumenti)

Servizi AWS

- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue risorse AWS e delle applicazioni che esegui su AWS in tempo reale.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.

- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito. Questa rete virtuale è simile a una rete tradizionale che gestiresti nel tuo data center, con i vantaggi dell'utilizzo dell'infrastruttura scalabile di AWS. Questo modello utilizza endpoint VPC e interfacce di rete elastiche.

Altri servizi

- HashiCorp Terraform è uno strumento open source di infrastruttura come codice (IaC) che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud. Questo modello utilizza Terraform per implementare l'infrastruttura.
- Puppeteer è una libreria Node.js. Il runtime CloudWatch Synthetics utilizza il framework Puppeteer.

Codice

La soluzione è disponibile nel repository cloud. GitHub watch-synthetics-canary-terraform Per ulteriori informazioni, consulta la sezione Informazioni aggiuntive.

Epiche

Implementa la soluzione per il monitoraggio di un URL privato

Operazione	Description	Competenze richieste
Raccogli i requisiti per il monitoraggio dell'URL privato.	Raccogli la definizione completa dell'URL: dominio, parametri e intestazioni. Per comunicare in privato con Amazon S3 e CloudWatch Amazon, utilizza gli endpoint VPC. Nota come il VPC e le sottoreti sono accessibili all'endpoint. Considerate la frequenza delle corse dei canarini.	Architetto del cloud, amministratore di rete

Operazione	Description	Competenze richieste
Modifica la soluzione esistente per monitorare l'URL privato.	Modifica il `terraform .tfvars` file: • `name`— Il nome del tuo canarino. • `runtime_version` — La versione runtime del canarino. Si consiglia di utilizzare syn-nodejs-puppeteer -3.7. • `take_screenshot` — Se è necessario scattare uno screenshot. • `api_hostname` — Il nome host dell'endpoint monitorato. • `api_path`— Il percorso dell'endpoint monitorato. • `vpc_id`— L'ID VPC utilizzato dalla funzione Lambda canary. • `subnet_ids` — La sottorete IDs utilizzata dalla funzione Lambda canary. • `frequency` — La frequenza di marcia del canarino in minuti. • `alert_sns_topic` — L'argomento SNS a cui viene inviata la notifica di CloudWatch allarme.	Architetto del cloud

Operazione	Description	Competenze richieste

Implementa e gestisci la soluzione.	Per distribuire la soluzione, procedi come segue:
	1. Dalla `cloudwatch-synthetics-canary-terraform` directory del tuo ambiente di sviluppo, inizializza Terraform.
	`terraform init`
	2. Pianifica e rivedi le modifiche.
	`terraform plan`
	3. Distribuire la soluzione.
	`terraform apply`
		Architetto del cloud, DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
L'eliminazione delle risorse assegnate si blocca.	Elimina manualmente la funzione Canary Lambda, l'elastic network interface corrispon dente e il gruppo di sicurezza, in quest'ordine.

Risorse correlate

- Utilizzo del monitoraggio sintetico
- Monitora gli endpoint API Gateway con Amazon CloudWatch Synthetics (post sul blog)

Informazioni aggiuntive

Artefatti del repository

Gli artefatti del repository hanno la seguente struttura.

```
.
### README.md
### main.tf
### modules
#   ### canary
#   ### canary-infra
### terraform.tfvars
### tf.plan
### variable.tf
```

Il `main.tf` file contiene il modulo principale e distribuisce due sottomoduli:

- `canary-infra`implementa l'infrastruttura necessaria per le isole canarie.
- `canary`dispiega i canarini.

I parametri di input per la soluzione si trovano nel `terraform.tfvars` file. È possibile utilizzare il seguente esempio di codice per creare un canarino.

```
module "canary" {
    source = "./modules/canary"
    name   = var.name
    runtime_version = var.runtime_version
    take_screenshot = var.take_screenshot
    api_hostname = var.api_hostname
    api_path = var.api_path
    reports-bucket = module.canary_infra.reports-bucket
    role = module.canary_infra.role
    security_group_id = module.canary_infra.security_group_id
    subnet_ids = var.subnet_ids
    frequency = var.frequency
    alert_sns_topic = var.alert_sns_topic
}
```

Segue il file.var corrispondente.

```
name   = "my-canary"
runtime_version = "syn-nodejs-puppeteer-3.7"
take_screenshot = false
api_hostname = "mydomain.internal"
api_path = "/path?param=value"
vpc_id = "vpc_id"
subnet_ids = ["subnet_id1"]
frequency = 5
alert_sns_topic = "arn:aws:sns:eu-central-1:111111111111:yyyyy"
```

Pulizia della soluzione

Se si esegue il test in un ambiente di sviluppo, è possibile ripulire la soluzione per evitare costi aggiuntivi.

1. Nella Console di gestione AWS, accedi alla console Amazon S3. Svuota il bucket Amazon S3 creato dalla soluzione. Assicurati di eseguire un backup dei dati, se necessario.
2. Nell'ambiente di sviluppo, dalla `cloudwatch-synthetics-canary-terraform` directory, esegui il `destroy` comando.

   ```
   terraform destroy
   ```

Implementa una ChatOps soluzione per gestire i risultati della scansione SAST utilizzando Amazon Q Developer nelle applicazioni di chat, azioni personalizzate e CloudFormation

Anand Bukkapatnam Tirumala, Amazon Web Services

Riepilogo

Questo modello presenta una soluzione completa che utilizza Amazon Q Developer nelle applicazioni di chat per semplificare la gestione degli errori di scansione statici dei test di sicurezza delle applicazioni (SAST) segnalati. SonarQube Questo approccio innovativo integra azioni e notifiche personalizzate in un'interfaccia conversazionale, abilitando processi decisionali e di collaborazione efficienti all'interno dei team di sviluppo.

Nell'odierno ambiente di sviluppo software frenetico, la gestione efficiente dei risultati di scansione SAST è fondamentale per mantenere la qualità e la sicurezza del codice. Tuttavia, molte organizzazioni devono affrontare le seguenti sfide significative:

- Ritardo nella consapevolezza delle vulnerabilità critiche a causa dell'inefficienza dei sistemi di notifica
- Processi decisionali lenti causati da flussi di lavoro di approvazione scollegati
- Mancanza di risposte immediate e attuabili agli errori di scansione SAST
- Comunicazione e collaborazione frammentate in merito ai problemi di sicurezza
- Configurazione manuale dell'infrastruttura per gli strumenti di sicurezza, dispendiosa in termini di tempo e soggetta a errori

Questi problemi spesso comportano un aumento dei rischi per la sicurezza, ritardi nei rilasci e una riduzione della produttività del team. Per affrontare efficacemente queste sfide è necessaria una soluzione in grado di semplificare la gestione dei risultati SAST, migliorare la collaborazione tra i team e automatizzare il provisioning dell'infrastruttura.

Le caratteristiche principali della soluzione includono:

- Notifiche personalizzate: gli avvisi e le notifiche in tempo reale vengono inviati direttamente ai canali di chat del team, garantendo una pronta consapevolezza e azioni in caso di vulnerabilità o errori della scansione SAST.

- Approvazioni conversazionali: le parti interessate possono avviare e completare i flussi di lavoro di approvazione dei risultati della scansione SAST senza problemi all'interno dell'interfaccia di chat, accelerando i processi decisionali.
- Azioni personalizzate: i team possono definire ed eseguire azioni personalizzate in base ai risultati della scansione SAST, come l'attivazione automatica di messaggi e-mail in caso di guasti ai gate di qualità, il miglioramento della reattività ai problemi di sicurezza.
- Collaborazione centralizzata: tutte le discussioni, le decisioni e le azioni relative alla scansione SAST vengono conservate all'interno di un ambiente di chat unificato, che favorisce una migliore collaborazione e condivisione delle conoscenze tra i membri del team.
- Infrastructure as code (IaC): l'intera soluzione è dotata di AWS CloudFormation modelli che consentono un provisioning dell'infrastruttura più rapido e affidabile riducendo al contempo gli errori di configurazione manuale.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Un ruolo AWS Identity and Access Management (IAM) con autorizzazioni per creare e gestire risorse associate a Servizi AWS quelle elencate in Strumenti.
- Uno spazio di lavoro Slack.
- Amazon Q Developer in applicazioni di chat aggiunte all'area di lavoro Slack richiesta come plug-in. Per ulteriori informazioni, consulta Aggiungere app al tuo spazio di lavoro Slack nella documentazione di Slack. Tieni nota dell'ID dell'area di lavoro Slack come mostrato dopo la registrazione avvenuta con successo. Console di gestione AWS
- Un client di applicazioni Amazon Q Developer in chat configurato, con l'ID dello spazio di lavoro immediatamente disponibile per l'immissione nella CloudFormation console. Per istruzioni, consulta Configurare un client Slack nella Guida per amministratori delle applicazioni Amazon Q Developer in chat.
- Un account e-mail di origine creato e verificato in Amazon Simple Email Service (Amazon SES) per inviare messaggi e-mail di approvazione. Per istruzioni di configurazione, consulta Creazione e verifica delle identità e-mail nella Amazon Simple Email Service Developer Guide.
- Un indirizzo e-mail di destinazione per ricevere le notifiche di approvazione. Questo indirizzo può essere una casella di posta condivisa o una lista di distribuzione specifica del team.

- Un' SonarQube istanza operativa accessibile dal tuo Account AWS. Per ulteriori informazioni, consulta le istruzioni SonarQube di installazione.
- Un token SonarQube utente con le autorizzazioni per attivare e creare progetti attraverso la pipeline.

Limitazioni

- La creazione di pulsanti di azione personalizzati è un processo manuale in questa soluzione.
- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Architecture

Il diagramma seguente mostra i componenti del flusso di lavoro e dell'architettura per questo modello.

Il diagramma mostra il flusso di lavoro automatizzato per il controllo automatico della qualità del codice:

1. Preparazione e caricamento del codice:
   - Lo sviluppatore comprime la codebase in un file.zip.
   - Lo sviluppatore carica manualmente il file.zip in un bucket Amazon Simple Storage Service (Amazon S3) designato.
2. Attivazione e orchestrazione di eventi Amazon S3: AWS Step Functions
   - L'evento di caricamento di Amazon S3 attiva un flusso di lavoro Step Functions.
   - Step Functions orchestra una scansione SAST utilizzando. SonarQube
   - Il flusso di lavoro monitora lo stato del AWS CodeBuild lavoro per determinare le azioni successive. In caso di CodeBuild esito positivo (quality gate pass), il flusso di lavoro viene interrotto. In caso di CodeBuild errore, viene richiamata una AWS Lambda funzione per la diagnostica. Per ulteriori dettagli, vedere la AWS Step Functions logica più avanti in questa sezione.
3. AWS CodeBuild esecuzione:
   - Il CodeBuild job esegue una SonarQube scansione sulla codebase caricata.

- Gli artefatti di scansione vengono archiviati in un bucket Amazon S3 separato per il controllo e l'analisi.

4. Analisi dei guasti (funzione Lambda):
   - In CodeBuild caso di errore, viene attivata la funzione `CheckBuildStatus` Lambda.
   - In caso di CodeBuild successo, il processo viene interrotto e non sono necessarie ulteriori azioni.
5. La funzione Lambda analizza la causa del guasto (guasto del Quality Gate o altri problemi)
   - La `CheckBuildStatus` funzione crea un payload personalizzato con informazioni dettagliate sugli errori.
   - La `CheckBuildStatus` funzione pubblica il payload personalizzato su un argomento di Amazon Simple Notification Service (Amazon SNS).
6. Sistema di notifica:
   - Amazon SNS inoltra il payload ad Amazon Q Developer nelle applicazioni di chat per l'integrazione con Slack.
7. Integrazione con Slack:
   - Amazon Q Developer nelle applicazioni di chat pubblica una notifica nel canale Slack designato.
8. Procedura di approvazione:
   - Gli approvatori esaminano i dettagli dell'errore nella notifica di Slack.
   - Gli approvatori possono avviare l'approvazione utilizzando il pulsante Approva in Slack.
9. Gestore dell'approvazione:
   - Una funzione Approval Lambda elabora l'azione di approvazione di Slack.
   - La funzione Approval pubblica il messaggio personalizzato su Amazon SES.
10. Messaggio generato:
    - La funzione di approvazione genera un messaggio personalizzato per la notifica agli sviluppatori.
11. Notifica per gli sviluppatori:
    - Amazon SES invia un messaggio e-mail allo sviluppatore con i passaggi successivi o le azioni richieste.

Questo flusso di lavoro combina il caricamento manuale del codice con controlli di qualità automatici, fornisce un feedback immediato tramite Slack e consente l'intervento umano quando necessario, garantendo un processo di revisione del codice solido e flessibile.

AWS Step Functions logica

Come mostrato nel diagramma di architettura precedente, se il quality gate pass on SonarQube fallisce, il flusso di lavoro passa alla funzione `CheckBuildStatus` Lambda. La `CheckBuildStatus` funzione attiva una notifica sul canale Slack. Ogni notifica include informazioni con i passaggi successivi suggeriti. Di seguito sono riportati i tipi di notifiche:

- L'applicazione non è riuscita nella scansione di sicurezza del codice: l'utente riceve questa notifica quando il codice caricato non ha superato la scansione SonarQube di sicurezza. L'utente può scegliere APPROVE per accettare la build. Tuttavia, la notifica consiglia all'utente di fare attenzione ai potenziali rischi di scarsa qualità del codice e alla sicurezza. La notifica include i seguenti dettagli:
  - Passaggi successivi: Errore: Quality gate status: FAILED — Visualizza i dettagli all'URL fornito.
  - Valuta le vulnerabilità come indicato nel documento all'URL fornito.
  - CodeBuild i dettagli sono disponibili nella posizione all'URL fornito.
- La pipeline di scansione dell'applicazione non è riuscita per qualche altro motivo: l'utente riceve questa notifica quando la pipeline non riesce per qualche motivo diverso dal fallimento della scansione di sicurezza del codice. La notifica include i seguenti dettagli:
  - Per i passaggi successivi, vai al link fornito per ulteriori informazioni sulla risoluzione dei problemi.

Per vedere gli screenshot delle notifiche così come appaiono in un canale Slack, vai alla cartella assets nell'archivio GitHub chatops-slack.

Il diagramma seguente mostra un esempio dello stato della fase di Step Functions dopo il fallimento del Quality Gate Pass.

Tools (Strumenti)

Servizi AWS

- Le applicazioni di chat di Amazon Q Developer ti consentono di utilizzare i canali di chat Amazon Chime, Microsoft Teams e Slack per monitorare e rispondere agli eventi operativi nelle tue applicazioni. AWS Avviso di fine del supporto: il 20 febbraio 2026, AWS terminerà il supporto per il servizio Amazon Chime. Dopo il 20 febbraio 2026, non potrai più accedere alla console Amazon

Chime o alle risorse dell'applicazione Amazon Chime. Per ulteriori informazioni, consulta il post del blog. Ciò non influisce sulla disponibilità del servizio Amazon Chime SDK.

- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita tra e. Account AWS Regioni AWS
- AWS CodeBuildè un servizio di compilazione completamente gestito che consente di compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per l'implementazione.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Con Gestione dei segreti AWS puoi sostituire le credenziali nel codice,incluse le password, con una chiamata API a Secrets Manager in modo da recuperare il segreto a livello di codice.
- Amazon Simple Email Service (Amazon SES) Simple Email Service (Amazon SES) ti aiuta a inviare e ricevere messaggi e-mail utilizzando i tuoi indirizzi e-mail e domini.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Step Functionsè un servizio di orchestrazione senza server che ti aiuta a combinare AWS Lambda funzioni e altro per creare applicazioni aziendali critiche. Servizi AWS

Altri strumenti

- Slack, un'offerta di Salesforce, è una piattaforma conversazionale basata sull'intelligenza artificiale che offre collaborazione tramite chat e video, automatizza i processi senza codice e supporta la condivisione delle informazioni.
- SonarQubeè uno strumento di analisi locale progettato per rilevare problemi di codifica in oltre 30 lingue, framework e piattaforme IaC.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub chatops-slack.

Best practice

- CloudFormation gestione dello stack: se riscontri errori durante l'esecuzione CloudFormation dello stack, ti consigliamo di eliminare lo stack fallito. Quindi, ricrealo con i valori dei parametri corretti. Questo approccio supporta una distribuzione pulita e aiuta a evitare potenziali conflitti o implementazioni parziali.
- Configurazione condivisa della posta in arrivo: quando configuri il `SharedInboxEmail` parametro, utilizza una lista di distribuzione comune accessibile a tutti gli sviluppatori interessati. Questo approccio promuove la trasparenza e aiuta le notifiche importanti a raggiungere i membri del team interessati.
- Flusso di lavoro di approvazione della produzione: per gli ambienti di produzione, limita l'accesso al canale Slack utilizzato per le approvazioni delle build. Solo gli approvatori designati devono essere membri di questo canale. Questa pratica mantiene una chiara catena di responsabilità e migliora la sicurezza limitando chi può approvare modifiche critiche.
- Autorizzazioni IAM: segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.

Epiche

Esegui la configurazione iniziale

Operazione	Description	Competenze richieste
Clonare il repository.	Per clonare il repository chatops-slack per questo pattern, usa il seguente comando. `git clone "git@github.com:aws-samples/chatops-slack.git"`	AWS DevOps, responsabile di sviluppo, DevOps ingegnere, amministratore cloud

Operazione	Description	Competenze richieste

Crea i file.zip che contengono il codice Lambda.	Crea i file.zip per il codice della AWS Lambda funzione and. `CheckBuildStatus` `ApprovalEmail` Per creare `notification.zip` `eapproval.zip` , utilizzate i seguenti comandi.
	`cd chatops-slack/src`
	`chmod -R 775 *`
	`zip -r approval.zip approval`
	`zip -r notification.zip notification`
		AWS DevOps, responsabile di sviluppo, DevOps ingegnere, amministratore cloud

Distribuisci il file stack pre-requisite.yml

Operazione	Description	Competenze richieste
Esegui il file `pre-requisite.yml` stack.	Il file `pre-requisite.yml` CloudFormation stack distribuisce le risorse iniziali necessarie prima di eseguire il `app-security.yml` file stack. Per eseguire il `pre-requisite.yml` file, effettuate le seguenti operazioni: 1. Accedere a Console di gestione AWS e aprire la	Amministratore AWS, AWS DevOps, responsabile di Build, DevOps ingegnere

Operazione	Description	Competenze richieste
	CloudFormationconsole. Scegli Crea stack, quindi scegli Con nuove risorse (standard) dall'elenco a discesa. 2. Nella pagina Crea stack, scegli Scegli un modello esistente e Carica un file modello. Quindi, scegli Scegli file e seleziona pre-requisite.yml. Scegli Next (Successivo). 3. Nella pagina Specifica re i dettagli dello stack, immettete i valori per i parametri come descritto in Informazioni aggiuntive. Quindi, seleziona Successiv o. 4. Nella pagina Configura le opzioni dello stack, scegli il ruolo IAM per la creazione delle risorse come descritto in Prerequisiti. Quindi, seleziona Successivo. 5. Nella pagina Rivedi e crea, scegli Invia. 6. Nella pagina dei dettagli dello stack, scegli le schede Risorse e Output. Prendete nota dei valori per i `CKMSKeyId` parametri `S3LambdaCKMSKeyArn` ,	

Operazione	Description	Competenze richieste
	e utilizzati nei passaggi seguenti.	
Carica i file.zip nel bucket Amazon S3.	Carica i `approval.zip` file `notification.zip` and che hai creato in precedenza nel bucket Amazon S3 denominato. `S3LambdaBucket` Il file `app-security.yml` CloudFormation stack viene utilizzato `S3LambdaBucket` per fornire la funzione Lambda.	AWS DevOps, responsabile di sviluppo, DevOps ingegnere, amministratore di sistema AWS

Esegui il file stack app-security.yml

Operazione	Description	Competenze richieste
Esegui il file `app-security.yml` stack.	Il file `app-security.yml` stack implementano l'infrastruttura rimanente per il sistema di notifica e approvazione. Per eseguire il `app-security.yml` file, procedi come segue: 1. Accedere a Console di gestione AWS e aprire la CloudFormationconsole. Scegli Crea stack, quindi scegli Con nuove risorse (standard) dall'elenco a discesa. 2. Nella pagina Crea stack, scegli Scegli un modello	AWS DevOps, amministratore di sistema AWS, DevOps ingegnere, responsabile Build

Operazione	Description	Competenze richieste
	esistente e Carica un file modello. Quindi, scegli Scegli file e seleziona app-security.yml. Scegli Next (Successivo). 3. Nella pagina Specificare i dettagli dello stack, inserisci i valori per i parametri come descritto in Informazioni aggiuntive. Quindi, seleziona Successivo. 4. Nella pagina Configura le opzioni dello stack, scegli il ruolo IAM per la creazione delle risorse come descritto in Prerequisiti. Quindi, seleziona Successivo. 5. Nella pagina Rivedi e crea, scegli Invia.	

Operazione	Description	Competenze richieste
Verifica la configurazione delle notifiche.	Per testare la configurazione delle notifiche, procedi come segue: 1. Apri la console Amazon SNS. Nel pannello di navigazione a sinistra, selezionare Topics (Argomenti). 2. Seleziona il nome dell'argomento che termina con LambdaToAWSSlackChatbot. 3. Nella pagina dei dettagli dell'argomento, scegli Pubblica messaggio. 4. Nella pagina Pubblica messaggio sull'argomento, per il corpo del messaggio da inviare all'endpoint, inserisci quanto segue: `{ "version": "1.0", "source": "custom", "content": { "description": ":warning: This is a test notification" } }` 5. Seleziona Publish message (Pubblica messaggio).	AWS DevOps, amministratore di sistema AWS, DevOps ingegnere, responsabile Build

Operazione	Description	Competenze richieste
	Dopo che il messaggio di prova è stato recapitato correttamente, dovresti vedere una notifica sul canale Slack. Per ulteriori informazioni, consulta Notifiche di test da Servizi AWS a Slack nella Guida per amministratori delle applicazioni Amazon Q Developer in chat.	

Imposta il flusso di approvazione

Operazione	Description	Competenze richieste
Configura un'azione Lambda personalizzata.	Per configurare l' AWS Lambda azione personali zzata, procedi come segue: 1. Scegli il pulsante con i puntini di sospensione verticali nella parte inferiore della notifica di consegna nel tuo canale Slack. 2. In Gestisci azioni, scegli Crea. 3. Inserisci un nome di azione personalizzato, ad esempio Approva. Questo nome è un identificatore univoco per l'azione personalizzata. 4. Inserisci un nome per il pulsante di azione personali zzato, ad esempio Approva.	Amministratore AWS, AWS DevOps, responsabile di Build, DevOps ingegnere, amministr atore Slack

Operazione	Description	Competenze richieste
	Questo nome viene visualizzato su un pulsante della notifica. Questo nome deve contenere al massimo 20 caratteri e può contenere emoji. 5. Per Tipo di azione personalizzato, selezionare Azione Lambda. 6. Scegli Next (Successivo). 7. Seleziona l'azione Account AWS e Regione AWS dove vuoi implementare questa azione. 8. Scegli Load Lambdas. 9. In Definisci funzione Lambda, seleziona una funzione Lambda che termina con. ApprovalEmailLambdaQuindi, seleziona Successivo. 10. Per creare il pulsante Approva, scegliete Salva nella pagina Criteri di visualizzazione.	

Operazione	Description	Competenze richieste
Convalida il flusso di approvazione.	Per verificare che il flusso di approvazione funzioni come previsto, scegli il pulsante Approva in Slack. Slackbot dovrebbe inviare una notifica sul thread dei messaggi con la stringa di conferma Email di approvazi one inviata con successo.	Amministratore AWS, AWS DevOps, DevOps ingegnere, amministratore Slack

risoluzione dei problemi

Problema	Soluzione
Configurazioni errate di Slack	Per informazioni sulla risoluzione dei problemi relativi alle configurazioni errate di Slack, consulta Troubleshooting Amazon Q Developer nella Guida per l'amministratore delle applicazi oni Amazon Q Developer in chat.
La scansione non è riuscita per qualche altro motivo	Questo errore indica che l'attività di compilazi one del codice non è riuscita. Per risolvere il problema, vai al link contenuto nel messaggio. L'errore dell'attività di compilazione del codice potrebbe avere le seguenti possibili cause: • L'applicazione non è impacchettata correttam ente. Il `sonar-scanner` comando non riesce a trovare il `sonar.project.env. properties` file. • I valori per i `SonarToken` parametri `SonarFileName` `SonarFileDirectory`

Problema	Soluzione
	, o non sono corretti. Controllate i valori, quindi eseguite nuovamente i file dello stack. • L'host Sonar non è raggiungibile. • Altri problemi per i quali è possibile risolvere utilizzando i registri.

Risorse correlate

AWS documentazione

- Configurare un client Slack
- Creazione di un'azione personalizzata
- Creazione di una procedura per l'identità di un indirizzo e-mail
- Tutorial: Inizia a usare Slack

Altre risorse

- Aggiungi app al tuo spazio di lavoro Slack (documentazione Slack)
- Generazione e utilizzo di token (documentazione) SonarQube
- Introduzione all'installazione del server (SonarQube documentazione)

Informazioni aggiuntive

Questa soluzione enfatizza le azioni personalizzate di Amazon Q Developer nelle applicazioni di chat per scopi di gestione dei rilasci. Tuttavia, puoi riutilizzare la soluzione modificando il codice Lambda per il tuo caso d'uso specifico e basandoti su di esso.

Parametri dei file stack CloudFormation

La tabella seguente mostra i parametri e le relative descrizioni per il file CloudFormation stack. `pre-requisite.yml`

Chiave	Descrizione

`StackName`	Il nome dello CloudFormation stack.
`S3LambdaBucket`	Il nome del bucket Amazon S3 in cui carichi il codice Lambda. Il nome deve essere univoco a livello globale.
`SonarToken`	Il token SonarQube utente come descritto in Prerequisiti.

La tabella seguente mostra i parametri e le relative descrizioni per il file CloudFormation stack. `app-security.yml`

Chiave	Descrizione
`CKMSKeyArn`	L' AWS KMS key Amazon Resource Name (ARN) utilizzato nei ruoli IAM e nelle funzioni Lambda create in questo stack.
`CKMSKeyId`	L' AWS KMS key ID utilizzato nell'argomento Amazon SNS creato in questo stack.
`EnvironmentType`	Il nome dell'ambiente client per la distribuzione della pipeline di scansione delle applicazioni. Seleziona il nome dell'ambiente dall'elenco a discesa dei valori consentiti.
`S3LambdaBucket`	Il nome del bucket Amazon S3 che contiene i `approval.zip` file and. `notification.zip`
`SESEmail`	Il nome dell'identità e-mail registrata in Amazon SES come descritto in Prerequisiti. Questa identità è l'indirizzo e-mail di origine.
`SharedInboxMail`	L'indirizzo e-mail di destinazione a cui vengono inviate le notifiche di scansione.

`SlackChannelId`	L'ID del canale Slack a cui desideri inviare le notifiche. Per trovare l'ID del canale, fai clic con il pulsante destro del mouse sul nome del canale in Dettagli del canale nell'app Slack. L'ID del canale si trova in basso.
`SlackWorkspaceId`	L'ID dell'area di lavoro di Slack come descritto in Prerequisiti. Per trovare l'ID dello spazio di lavoro Slack, accedi a Console di gestione AWS, apri la console delle applicazioni di chat Amazon Q Developer e scegli Clienti configura ti, Slack, WorkspaceID.
`StackName`	Il nome dello stack. CloudFormation
`SonarFileDirectory`	La directory che contiene il `sonar.pro ject.<env>.properties` file.
`SonarFileName`	Il nome del `sonar.project.<env >properties` file.
`SourceCodeZip`	Il nome del file.zip che contiene il `sonar.pro ject.<env>properties` file e il codice sorgente.

Implementa sistemi agentici su Amazon Bedrock con il framework CrewAI utilizzando Terraform

Vanitha Dontireddy, Amazon Web Services

Riepilogo

Questo modello dimostra come implementare sistemi di intelligenza artificiale scalabili multiagente utilizzando il framework CrewAI integrato con Amazon Bedrock e Terraform. La soluzione consente alle organizzazioni di creare, implementare e gestire sofisticati flussi di lavoro di agenti AI tramite l'infrastruttura come codice (IaC). In questo modello, le funzionalità di orchestrazione multiagente di CrewAI si combinano con i modelli di base di Amazon Bedrock e l'automazione dell'infrastruttura Terraform. Di conseguenza, i team possono creare sistemi di intelligenza artificiale pronti per la produzione che affrontano attività complesse con una supervisione umana minima. Il modello implementa sicurezza, scalabilità e best practice operative di livello aziendale.

Prerequisiti e limitazioni

Prerequisiti

- Un dispositivo attivo Account AWS con le autorizzazioni appropriate per accedere ai modelli Amazon Bedrock Foundation
- È installata la versione 1.5 o successiva di Terraform
- Python versione 3.9 o successiva installata
- Framework CrewAI installato

Limitazioni

- Le interazioni con gli agenti sono limitate dalle finestre di contesto del modello.
- A questo modello si applicano le considerazioni sulla gestione dello stato di Terraform per le implementazioni su larga scala.
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per informazioni sulla disponibilità per regione, vedi AWS Servizi per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Architecture

In questo schema, si verificano le seguenti interazioni:

- Amazon Bedrock fornisce le basi per l'intelligence degli agenti attraverso la sua suite di modelli di base (FMs). Abilita l'elaborazione del linguaggio naturale (NLP), il ragionamento e le capacità decisionali per gli agenti di intelligenza artificiale, mantenendo al contempo disponibilità e scalabilità elevate.
- Il framework CrewAI funge da livello di orchestrazione principale per la creazione e la gestione di agenti AI. Gestisce i protocolli di comunicazione degli agenti, la delega delle attività e la gestione del flusso di lavoro integrandosi con Amazon Bedrock.
- Terraform gestisce l'intero stack di infrastruttura tramite codice, incluse risorse di calcolo, reti, gruppi di sicurezza e AWS Identity and Access Management ruoli (IAM). Garantisce implementazioni coerenti e controllate dalla versione in tutti gli ambienti. L'implementazione di Terraform crea quanto segue:
  - AWS Lambda funzione per eseguire l'applicazione CrewAI
  - Bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) per codice e report
  - Ruoli IAM con autorizzazioni appropriate
  - CloudWatch Registrazione su Amazon
  - Esecuzione programmata da Amazon EventBridge

Il diagramma seguente illustra l'architettura per la distribuzione di sistemi multiagente CrewAI utilizzando Amazon Bedrock e Terraform.

Il diagramma mostra il flusso di lavoro seguente:

1. L'utente clona il repository.
2. L'utente esegue il comando `terraform apply` per distribuire le risorse. AWS
3. La configurazione del modello Amazon Bedrock include la specifica del modello di base (FM) da utilizzare per configurare gli agenti CrewAI.
4. Viene stabilita una EventBridge regola per attivare la funzione Lambda in base alla pianificazione definita.

5. Quando viene attivata (tramite pianificazione o manualmente), la funzione Lambda inizializza e assume il ruolo IAM con autorizzazioni di accesso e Amazon Bedrock. Servizi AWS
6. Il framework CrewAI carica le configurazioni degli agenti dai file YAML e crea agenti AI specializzati (il team di controllo della sicurezza dell'infrastruttura).AWS La funzione Lambda esegue in sequenza questi agenti per scansionare AWS le risorse, analizzare le vulnerabilità di sicurezza e generare report di controllo completi.
7. CloudWatch Logs acquisisce informazioni dettagliate sull'esecuzione dalla funzione Lambda con un periodo di conservazione di 365 giorni e crittografia AWS Key Management Service ()AWS KMS per i requisiti di conformità. I log forniscono visibilità sulle attività degli agenti, sul monitoraggio degli errori e sulle metriche delle prestazioni, consentendo un monitoraggio e una risoluzione dei problemi efficaci del processo di controllo di sicurezza.
8. Il rapporto di controllo di sicurezza viene generato automaticamente e archiviato nel bucket Amazon S3 designato. La configurazione automatizzata aiuta a mantenere un monitoraggio costante della sicurezza con un sovraccarico operativo minimo.

Dopo l'implementazione iniziale, il flusso di lavoro fornisce controlli e report di sicurezza continui per l' AWS infrastruttura senza intervento manuale.

Panoramica degli agenti AI

Questo modello crea più agenti di intelligenza artificiale, ciascuno con ruoli, obiettivi e strumenti unici:

- L'agente analista della sicurezza raccoglie e analizza AWS le informazioni sulle risorse.
- L'agente penetration tester identifica le vulnerabilità nelle risorse. AWS
- L'agente esperto di conformità verifica le configurazioni rispetto agli standard di conformità.
- L'agente che redige i report compila i risultati in report completi.

Questi agenti collaborano su una serie di attività, sfruttando le proprie competenze collettive per eseguire controlli di sicurezza e generare report completi. (Il `config/agents.yaml` file descrive le capacità e le configurazioni di ciascun agente di questo team.)

L'elaborazione dell'analisi della sicurezza consiste nelle seguenti azioni:

1. L'agente di analisi della sicurezza esamina i dati raccolti su AWS risorse come le seguenti:
   - Istanze e gruppi di sicurezza Amazon Elastic Compute Cloud (Amazon EC2)
   - Bucket e configurazioni Amazon S3

- Ruoli, politiche e autorizzazioni IAM
- Configurazioni e impostazioni di rete del cloud privato virtuale (VPC)
- Database e impostazioni di sicurezza di Amazon RDS
- Funzioni e configurazioni Lambda
- Altro Servizi AWS nell'ambito dell'audit

2. L'agente del penetration tester identifica potenziali vulnerabilità.
3. Gli agenti collaborano tramite il framework CrewAI per condividere i risultati.

La generazione di report consiste nelle seguenti azioni:

1. L'agente autore del report raccoglie i risultati di tutti gli altri agenti.
2. I problemi di sicurezza sono organizzati in base al servizio, alla gravità e all'impatto sulla conformità.
3. Per ogni problema identificato vengono generati consigli per la correzione.
4. Un report completo di audit di sicurezza viene creato in formato markdown e caricato nel bucket Amazon S3 designato. I report storici vengono conservati per il monitoraggio della conformità e il miglioramento del livello di sicurezza.

Le attività di registrazione e monitoraggio includono:

- CloudWatch i registri registrano i dettagli di esecuzione e gli eventuali errori.
- Le metriche di esecuzione Lambda vengono registrate per il monitoraggio.

> **Note**
>
> Il codice per `aws-security-auditor-crew` proviene dal repository GitHub 3P-Agentic_Frameworks, disponibile nella raccolta Samples. AWS

Disponibilità e scalabilità

È possibile espandere gli agenti disponibili a più di quattro agenti principali. Per ampliare la gamma con agenti specializzati aggiuntivi, prendete in considerazione i seguenti nuovi tipi di agenti:

- Un agente specializzato in intelligence sulle minacce può fare quanto segue:

    - Monitora i feed di minacce esterne e li correla con i risultati interni
    - Fornisce un contesto sulle minacce emergenti rilevanti per l'infrastruttura
    - Assegna priorità alle vulnerabilità in base allo sfruttamento attivo in ambiente naturale
- Gli agenti del framework di conformità possono concentrarsi su aree normative specifiche come le seguenti:
    - Agente di conformità allo standard PCI DSS (Payment Card Industry Data Security Standard)
    - Responsabile della conformità all'Health Insurance Portability and Accountability Act del 1996 (HIPAA)
    - Agente di conformità di System and Organization Controls 2 (SOC 2)
    - Responsabile della conformità al Regolamento generale sulla protezione dei dati (GDPR)

Ampliando attentamente gli agenti disponibili, questa soluzione può fornire informazioni sulla sicurezza più approfondite e specializzate, mantenendo al contempo la scalabilità in ambienti di grandi dimensioni. AWS Per ulteriori informazioni sull'approccio di implementazione, sullo sviluppo degli strumenti e sulle considerazioni sulla scalabilità, consulta Informazioni aggiuntive.

Tools (Strumenti)

Servizi AWS

- Amazon Bedrock è un servizio di intelligenza artificiale completamente gestito che rende disponibili modelli di base ad alte prestazioni (FMs) per l'uso tramite un'API unificata.
- Amazon CloudWatch Logs ti aiuta a centralizzare i log di tutti i tuoi sistemi e applicazioni, Servizi AWS così puoi monitorarli e archiviarli in modo sicuro.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altro modo. Account AWS In questo modello, viene utilizzato per pianificare e orchestrare i flussi di lavoro degli agenti.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.

- AWS SDK per Python (Boto3)è un kit di sviluppo software che ti aiuta a integrare la tua applicazione, libreria o script Python con. Servizi AWS
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati. In questo modello, fornisce l'archiviazione degli oggetti per gli artefatti degli agenti e la gestione dello stato.

Altri strumenti

- CrewAI è un framework open source basato su Python per la creazione di sistemi di intelligenza artificiale multiagente.
- Terraform è uno strumento di infrastruttura come codice (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.

Deposito di codici

Il codice per questo pattern è disponibile nel GitHub deploy-crewai-agents-terraformrepository.

Best practice

- Implementa una corretta gestione dello stato per Terraform utilizzando un backend Amazon S3 con blocco Amazon DynamoDB. Per ulteriori informazioni, consulta le migliori pratiche di backend in Le migliori pratiche per l'utilizzo del provider Terraform. AWS
- Usa gli spazi di lavoro per separare gli ambienti di sviluppo, allestimento e produzione.
- Segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.
- Abilita la registrazione e il monitoraggio dettagliati tramite CloudWatch Logs.
- Implementa meccanismi di ripetizione dei tentativi e gestione degli errori per le operazioni degli agenti.

Epiche

Implementa il framework CrewAI

Operazione	Description	Competenze richieste

Clonare il repository.	Per clonare l'archivio di questo pattern sul computer locale, esegui il seguente comando:
	`git clone "git@github.com:aws-samples/deploy-crewai-agents-terraform.git"`
	`cd deploy-crewai-agents-terraform`
		DevOps ingegnere

Modifica le variabili di ambiente.	Per modificare le variabili di ambiente, procedi come segue: 1. Crea un `terraform.tfvars` file dall'esempio `terraform.tfvars.example` nella directory terraform. 2. Modifica le variabili di ambiente per utilizzare le tue informazioni.	DevOps ingegnere

Crea l'infrastruttura.	Per creare l'infrastruttura, esegui i seguenti comandi:
	`cd terraform`
	`terraform init`
		DevOps ingegnere

Operazione	Description	Competenze richieste

	`terraform plan`
	
	Esamina attentamente il piano di esecuzione. Se le modifiche pianificate sono accettabili, esegui il seguente comando:
	
	`terraform apply --auto-approve`	

Accedi agli agenti CrewAI

Operazione	Description	Competenze richieste
Accedi agli agenti.	Gli agenti del team di AWS Infrastructure Security Audit and Reporting vengono implementati come funzione Lambda. Per accedere agli agenti, procedi nel seguente modo: 1. Accedi a Console di gestione AWS e apri la AWS Lambda console all'indirizzo https://console.aws.amazon.com/lambda/. 2. Nella pagina Funzioni, trova e seleziona la funzione denominata`{project_name} \` function (come definita nelle variabili Terraform).	DevOps ingegnere

Operazione	Description	Competenze richieste
	3. Dalla pagina della funzione, puoi eseguire le seguenti azioni: • Visualizzare i dettagli della configurazione. • Monitora le metriche di esecuzione. • Visualizza i CloudWatch registri. • Testare la funzione manualmente.	

Operazione	Description	Competenze richieste
(Facoltativo) Configurare l'esecuzione manuale degli agenti.	Gli agenti sono configurati per essere eseguiti automatic amente in base a una pianifica zione giornaliera (mezzanot te UTC). Tuttavia, è possibile attivarli manualmente utilizzan do i seguenti passaggi: 1. Nella console Lambda, seleziona la funzione denominata. `{project_ name}` 2. Nella pagina della funzione, scegli la scheda Test. 3. Crea un nuovo evento di test con un oggetto `{}` JSON vuoto. 4. Per eseguire l'evento, scegli Test. Per maggiori dettagli, consulta Testare le funzioni Lambda nella console nella documenta zione di Lambda.	DevOps ingegnere

Operazione	Description	Competenze richieste
Accedi ai log degli agenti per il debug.	Gli agenti CrewAI funzionano in un ambiente Lambda con le autorizzazioni necessarie per eseguire controlli di sicurezza e archiviare report in Amazon S3. L'output è un rapporto di markdown che fornisce un'analisi completa della sicurezza dell'infrastruttura. AWS Per facilitare il debug dettagliato del comportamento degli agenti, procedi come segue: 1. In, accedi a Console di gestione AWS Registri. CloudWatch 2. Trova il gruppo di log per la tua funzione Lambda. 3. Cerca le voci di registro con i nomi degli agenti (ad esempio, specialista di mappatura dell'infrastruttura e analista della sicurezza Exploratory). 4. Esamina i log per informazioni dettagliate sulle azioni di ciascun agente.	DevOps ingegnere

Operazione	Description	Competenze richieste
Visualizza i risultati dell'esecuzione dell'agente.	Per visualizzare i risultati dell'esecuzione di un agente, effettuate le seguenti operazioni: 1. Nel Console di gestione AWS, accedi ad Amazon S3. 2. Apri il bucket Amazon S3 denominato `{project_name}-reports-{random_suffix}` (come definito nelle variabili Terraform). I report vengono archiviati con nomi di file basati su timestamp come segue: `security-audit-report-YYYY-MM-DD-HH-MM-SS.md`)	DevOps ingegnere

Operazione	Description	Competenze richieste
Monitora l'esecuzione degli agenti.	Per monitorare l'esecuzione degli agenti tramite i CloudWatch log, effettuate le seguenti operazioni: 1. In, accedete Console di gestione AWS a. CloudWatch 2. Vai a Gruppi di log. 3. Seleziona il gruppo di log denominato `/aws/lambda/{project_name}-function` . 4. In Log stream, scegli il flusso di log più recente per visualizzare informazioni dettagliate sull'esecuzione.	DevOps ingegnere

Operazione	Description	Competenze richieste

Personalizza il comportamento degli agenti.	Per modificare gli agenti o le loro attività, procedi come segue:
	1. Aggiorna i file di configurazione nel tuo repository locale:
	• Il file seguente definisce il ruolo, le funzionalità e le impostazioni di ciascun agente: `src/aws_infrastructure_security_audit_and_reporting/config/agents.yaml`
	• Il file seguente definisce le attività e i flussi di lavoro per gli agenti:
	`src/aws_infrastructure_security_audit_and_reporting/config/tasks.yaml`
	2. Per riconfezionare e aggiornare la funzione Lambda, usa i seguenti comandi:
	`cd terraform`
	`terraform apply`
		DevOps ingegnere

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste

Eliminare le risorse create.	Per eliminare tutta l'infrast ruttura creata da questo modello, esegui il seguente comando:
	`terraform plan -destroy`
	Warning
	Il comando seguente eliminerà definitiv amente tutte le risorse create da questo pattern. Il comando richiederà una conferma prima di rimuovere qualsiasi risorsa.
	Esamina attentamente il piano di distruzione. Se le eliminazi oni pianificate sono accettabili, esegui il seguente comando:
	`terraform destroy`
		DevOps ingegnere

Risoluzione dei problemi

Problema	Soluzione
Comportamento dell'agente	Per informazioni su questo problema, consulta Test e risoluzione dei problemi del comportamento degli agenti nella documentazione di Amazon Bedrock.
Problemi relativi alla rete Lambda	Per informazioni su questi problemi, consulta Risoluzione dei problemi di rete in Lambda nella documentazione di Lambda.
autorizzazioni IAM	Per informazioni su questi problemi, consulta Risoluzione dei problemi di IAM nella documentazione IAM.

Risorse correlate

AWS Blog

- Crea sistemi agentici con CrewAI e Amazon Bedrock

AWS documentazione

- Documentazione Amazon Bedrock
- Come funziona Amazon Bedrock Agents
- AWS Well-Architected Framework

Altre risorse

- Documentazione CrewAI
- Documentazione Terraform Provider AWS

Informazioni aggiuntive

Questa sezione contiene informazioni su un approccio di implementazione, sullo sviluppo di strumenti e su considerazioni sulla scalabilità relative alla discussione precedente in Automazione e scalabilità.

Approccio all'implementazione

Considerate il seguente approccio all'aggiunta di agenti:

1. Configurazione dell'agente:
   - Aggiungere nuove definizioni degli agenti al `config/agents.yaml` file.
   - Definisci retroscena, obiettivi e strumenti specializzati per ogni agente.
   - Configura le funzionalità di memoria e analisi in base alla specializzazione dell'agente.
2. Orchestrazione delle attività:
   - Aggiorna il `config/tasks.yaml` file per includere nuove attività specifiche per l'agente.
   - Crea dipendenze tra le attività per garantire un flusso di informazioni adeguato.
   - Implementare l'esecuzione parallela delle attività laddove appropriato.

Implementazione tecnica

Di seguito è riportata un'aggiunta al `agents.yaml` file relativo a un agente Threat Intelligence Specialist proposto:

```
Example new agent configuration in agents.yaml
threat_intelligence_agent:
 name: "Threat Intelligence Specialist"
 role: "Cybersecurity Threat Intelligence Analyst"
 goal: "Correlate AWS security findings with external threat intelligence"
 backstory: "Expert in threat intelligence with experience in identifying emerging
 threats and attack patterns relevant to cloud infrastructure."
verbose: true
allow_delegation: true
tools:
- "ThreatIntelligenceTool"
- "AWSResourceAnalyzer"
```

Sviluppo di strumenti

Con il framework CrewAI, puoi intraprendere le seguenti azioni per migliorare l'efficacia del personale addetto ai controlli di sicurezza:

- Crea strumenti personalizzati per i nuovi agenti.
- Integrazione con sistemi esterni APIs per l'intelligence sulle minacce.
- Sviluppa analizzatori specializzati per diversi tipi di applicazioni. Servizi AWS

Considerazioni sulla scalabilità

Quando espandi il tuo sistema di audit e reporting sulla sicurezza AWS dell'infrastruttura per gestire ambienti più grandi o audit più completi, tieni conto dei seguenti fattori di scalabilità:

- Risorse computazionali
  - Aumenta l'allocazione della memoria Lambda per gestire agenti aggiuntivi.
  - Prendi in considerazione la possibilità di suddividere i carichi di lavoro degli agenti tra più funzioni Lambda.
- Gestione dei costi
  - Monitora l'utilizzo dell'API Amazon Bedrock man mano che aumenta il numero di agenti.
  - Implementa l'attivazione selettiva degli agenti in base all'ambito dell'audit.
- Efficienza della collaborazione
  - Ottimizza la condivisione delle informazioni tra gli agenti.
  - Implementa strutture gerarchiche di agenti per ambienti complessi.
- Miglioramento della base di conoscenze
  - Fornisci agli agenti basi di conoscenza specializzate per i loro domini.
  - Aggiorna regolarmente le conoscenze degli agenti con nuove best practice di sicurezza.

Implementa un job AWS Glue con una pipeline CodePipeline CI/CD AWS

Bruno Klein e Luis Henrique Massao Yamada, Amazon Web Services

Riepilogo

Questo modello dimostra come integrare AWS CodeCommit e AWS CodePipeline con AWS Glue e utilizzare AWS Lambda per avviare lavori non appena uno sviluppatore invia le modifiche a un repository AWS remoto. CodeCommit

Quando uno sviluppatore invia una modifica a un repository di estrazione, trasformazione e caricamento (ETL) e invia le modifiche ad AWS CodeCommit, viene richiamata una nuova pipeline. La pipeline avvia una funzione Lambda che avvia un job AWS Glue con queste modifiche. Il job AWS Glue esegue il task ETL.

Questa soluzione è utile nel caso in cui aziende, sviluppatori e ingegneri dei dati vogliano avviare attività non appena le modifiche vengono apportate e trasferite negli archivi di destinazione. Aiuta a raggiungere un livello più elevato di automazione e riproducibilità, evitando quindi errori durante l'avvio e il ciclo di vita del lavoro.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Git installato sul computer locale
- Amazon Cloud Development Kit (Amazon CDK) installato sul computer locale
- Python installato sulla macchina locale
- Il codice nella sezione Allegati

Limitazioni

- La pipeline è completata non appena il job AWS Glue viene avviato con successo. Non aspetta la conclusione del lavoro.
- Il codice fornito nell'allegato è destinato esclusivamente a scopi dimostrativi.

Architecture

Stack tecnologico Target

- AWS Glue
- AWS Lambda
- AWS CodePipeline
- AWS CodeCommit

Architettura Target

Il processo consiste nei seguenti passaggi:

1. Lo sviluppatore o l'ingegnere dei dati apporta una modifica al codice ETL, esegue il commit e invia la modifica ad AWS. CodeCommit
2. Il push avvia la pipeline.
3. La pipeline avvia una funzione Lambda, che richiama il repository e carica il file `codecommit:GetFile` su Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3).
4. La funzione Lambda avvia un nuovo job AWS Glue con il codice ETL.
5. La funzione Lambda completa la pipeline.

Automazione e scalabilità

L'allegato di esempio dimostra come integrare AWS Glue con AWS CodePipeline. Fornisce un esempio di base che puoi personalizzare o estendere per uso personale. Per i dettagli, consulta la sezione Epics.

Tools (Strumenti)

- AWS CodePipeline: AWS CodePipeline è un servizio di distribuzione continua completamente gestito che ti aiuta ad automatizzare le pipeline di rilascio per aggiornamenti rapidi e affidabili di applicazioni e infrastrutture.
- AWS CodeCommit: AWS CodeCommit è un servizio di controllo del codice sorgente completamente gestito che ospita repository sicuri basati su Git.

- AWS Lambda — AWS Lambda è un servizio di elaborazione serverless che consente di eseguire codice senza effettuare il provisioning o la gestione di server.
- AWS Glue — AWS Glue è un servizio di integrazione dei dati senza server che semplifica la scoperta, la preparazione e la combinazione di dati per l'analisi, l'apprendimento automatico e lo sviluppo di applicazioni.
- Client Git: Git fornisce strumenti GUI oppure puoi usare la riga di comando o uno strumento desktop per controllare gli artefatti richiesti. GitHub
- AWS CDK: AWS CDK è un framework di sviluppo software open source che ti aiuta a definire le risorse delle tue applicazioni cloud utilizzando linguaggi di programmazione familiari.

Epiche

Distribuisci il codice di esempio

Operazione	Description	Competenze richieste
Configurare .	Configura l'AWS Command Line Interface (AWS CLI) per effettuare il targeting e l'autenticazione con il tuo account AWS corrente. Per istruzioni, consulta la documentazione dell'interfaccia a riga di comando di AWS.	Sviluppatore, DevOps ingegnere
Estrai i file di progetto di esempio.	Estrai i file dall'allegato per creare una cartella contenente i file di progetto di esempio.	Sviluppatore, DevOps ingegnere

Implementa il codice di esempio.	Dopo aver estratto i file, esegui i seguenti comandi dalla posizione di estrazione per creare un esempio di base:
	`cdk bootstrap`
	`cdk deploy`
		Sviluppatore, DevOps ingegnere

Operazione	Description	Competenze richieste

	`git init`
	`git remote add origin <code-commit-repository-url>`
	`git stage .`
	`git commit -m "adds sample code"`
	`git push --set-upstream origin main`
	
	Dopo l'ultimo comando, puoi monitorare lo stato della pipeline e del job AWS Glue.	

Personalizza il codice.	Personalizzate il codice per il file etl.py in base ai vostri requisiti aziendali. È possibile rivedere il codice ETL, modificare le fasi della pipeline o estendere la soluzione.	Ingegnere dei dati

Risorse correlate

- Guida introduttiva alla CDK AWS
- Aggiungere lavori in AWS Glue
- Integrazioni Source Action in CodePipeline
- Richiama una funzione AWS Lambda in una pipeline in CodePipeline
- Programmazione AWS Glue
- CodeCommit GetFile API AWS

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Distribuisci codice in più regioni AWS utilizzando AWS CodePipeline CodeCommit, AWS e AWS CodeBuild

Anand Krishna Varanasi, Amazon Web Services

Riepilogo

Questo modello dimostra come creare un'infrastruttura o un'architettura in più regioni di Amazon Web Services (AWS) utilizzando AWS CloudFormation. Include integrazione continua (CI) /distribuzione continua (CD) in più regioni AWS per implementazioni più rapide. I passaggi di questo modello sono stati testati per la creazione di un CodePipeline lavoro AWS da distribuire in tre regioni AWS, ad esempio. Puoi modificare il numero di regioni in base al tuo caso d'uso.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- • Un CodeBuild ruolo con AmazonS3 FullAccess e le politiche. CloudWatchFullAccess Queste policy consentono di CodeBuild guardare gli eventi di AWS CodeCommit tramite Amazon CloudWatch e di utilizzare Amazon Simple Storage Service (Amazon S3) come archivio di artefatti.
  - Un CloudFormation ruolo AWS con le seguenti policy, che offrono ad AWS CloudFormation, nella fase finale di build, la possibilità di creare o aggiornare funzioni AWS Lambda, inviare o guardare CloudWatch i log di Amazon e creare e aggiornare set di modifiche.
    - AWSLambdaFullAccess
    - AWSCodeDeployFullAccess
    - CloudWatchFullAccess
    - AWSCloudFormationFullAccess
    - AWSCodePipelineFullAccess

> **Note**
>
> Due ruoli AWS Identity and Access Management (IAM) per AWS CodeBuild e AWS CloudFormation con policy adeguate CodeBuild per eseguire le attività CI di test, raggruppamento, imballaggio degli artefatti e distribuzione in più regioni AWS in parallelo.

Verifica incrociando le policy create da CodePipeline per verificare che CodeBuild AWS CloudFormation disponga delle autorizzazioni appropriate nelle fasi CI e CD.

Architecture

L'architettura e il flusso di lavoro multiregione di questo modello comprendono i seguenti passaggi.

1. Il codice viene inviato a un repository. CodeCommit
2. Dopo aver ricevuto un aggiornamento o un commit del codice, CodeCommit richiama un CloudWatch evento, che a sua volta avvia un processo. CodePipeline
3. CodePipeline coinvolge il CI gestito da. CodeBuild Vengono eseguite le seguenti attività.
   - Test dei CloudFormation modelli AWS (opzionale)
   - Pacchetto dei CloudFormation modelli AWS per ogni regione inclusa nella distribuzione. Ad esempio, questo modello viene distribuito in parallelo su tre regioni AWS, quindi raggruppa CodeBuild i CloudFormation modelli AWS in tre bucket S3, uno in ciascuna regione specificata. I bucket S3 vengono utilizzati solo come repository di artefatti CodeBuild .
4. CodeBuild impacchetta gli artefatti come input per la fase successiva di distribuzione, che viene eseguita in parallelo nelle tre regioni AWS. Se specifichi un numero diverso di regioni, CodePipeline verrà distribuito in tali regioni.

Tools (Strumenti)

Strumenti

- AWS CodePipeline: CodePipeline è un servizio di distribuzione continua che puoi utilizzare per modellare, visualizzare e automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- AWS CodeBuild: CodeBuild è un servizio di build completamente gestito che compila il codice sorgente, esegue test unitari e produce artefatti pronti per la distribuzione.
- AWS CodeCommit: CodeCommit è un servizio di controllo delle versioni ospitato da Amazon Web Services che puoi utilizzare per archiviare e gestire in modo privato risorse (come codice sorgente e file binari) nel cloud.

- AWS CloudFormation: AWS CloudFormation è un servizio che ti aiuta a modellare e configurare le tue risorse Amazon Web Services in modo da poter dedicare meno tempo alla gestione di tali risorse e più tempo a concentrarti sulle applicazioni eseguite in AWS.
- AWS Identity and Access Management — AWS Identity and Access Management (IAM) è un servizio Web che ti aiuta a controllare in modo sicuro l'accesso alle risorse AWS.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è uno storage per Internet. È concepito per rendere più accessibili agli sviluppatori risorse informatiche su grande scala per il Web.

Codice

Il seguente codice di esempio è per il `BuildSpec.yaml` file (fase di compilazione).

```
---
artifacts:
discard-paths: true
files:
- packaged-first-region.yaml
- packaged-second-region.yaml
- packaged-third-region.yaml
phases:
build:
commands:
- echo "********BUILD PHASE - CF PACKAGING**********"
- "aws cloudformation package --template-file sam-template.yaml --s3-bucket
 $S3_FIRST_REGION --output-template-file packaged-first-region.yaml --region
 $FIRST_REGION"
- "aws cloudformation package --template-file sam-template.yaml --s3-bucket
 $S3_SECOND_REGION --output-template-file packaged-second-region.yaml --region
 $SECOND_REGION"
- "aws cloudformation package --template-file sam-template-anand.yaml --s3-bucket
 $S3_THIRD_REGION --output-template-file packaged-third-region.yaml --region
 $THIRD_REGION"
install:
commands:
- echo "********BUILD PHASE - PYTHON SETUP**********"
runtime-versions:
python: 3.8
post_build:
commands:
- echo "********BUILD PHASE - PACKAGING COMPLETION**********"
```

```
pre_build:
commands:
- echo "********BUILD PHASE - DEPENDENCY SETUP**********"
- "npm install --silent --no-progress"
- echo "********BUILD PHASE - DEPENDENCY SETUP DONE**********"
version: 0.2
```

Epiche

Prepara il codice e il repository CodeCommit

Operazione	Description	Competenze richieste
Seleziona la regione AWS principale per la distribuzione.	Accedi al tuo account AWS e scegli la regione principale per la distribuzione. Il CodeCommit repository si troverà nella regione principale.	DevOps
Crea il CodeCommit repository.	Crea il CodeCommit repository e inserisci il codice richiesto. Il codice include generalmente i modelli AWS CloudFormation o AWS SAM, l'eventuale codice Lambda e i CodeBuild `buildspec.yaml` file come input per AWS. CodePipeline	DevOps
Inserisci il codice nel CodeCommit repository.	Nella sezione Allegati, scarica il codice per questo esempio, quindi inserisci il codice richiesto. In genere, il codice può includere modelli AWS CloudFormation o AWS SAM, codice Lambda e CodeBuild `buildspec.yaml` file come input per la pipeline.	DevOps

Fase di origine: creazione della pipeline

Operazione	Description	Competenze richieste
Crea il CodePipeline lavoro.	Sulla CodePipeline console, scegli Crea pipeline.	DevOps
Assegna un nome al CodePipeline lavoro e scegli l'impostazione del ruolo di servizio.	Inserisci un nome per il lavoro e mantieni l'impostazione predefinita del ruolo di servizio in modo da CodePipeline creare il ruolo con le politiche necessarie allegate.	DevOps
Specificate la posizione del deposito degli artefatti.	In Impostazioni avanzate, mantieni l'opzione predefini ta in modo da CodePipel ine creare un bucket S3 da utilizzare per l'archiviazione degli artefatti del codice. Se invece utilizzi un bucket S3 esistente, il bucket deve trovarsi nella regione principal e che hai specificato nella prima epic.	DevOps
Specificate la chiave di crittografia.	Mantieni l'opzione predefini ta, Default AWS Managed Key, o scegli di utilizzare la tua chiave gestita dal cliente AWS Key Management Service (AWS KMS).	DevOps
Specificare il provider di origine.	In Source provider, scegli AWS CodeCommit.	DevOps
Specificate il repository.	Scegli il CodeCommit repository che hai creato nella	DevOps

Operazione	Description	Competenze richieste
	prima epopea. Se hai inserito il codice in un ramo, scegli il ramo.	
Specificate come vengono rilevate le modifiche al codice.	Mantieni l'impostazione predefinita, Amazon CloudWatch Events, come trigger di modifica CodeCommit per avviare il CodePipeline processo.	DevOps

Fase di costruzione: configura la pipeline

Operazione	Description	Competenze richieste
Specificare il fornitore della build.	Per il fornitore di build, scegli AWS CodeBuild.	DevOps
Specificare la regione AWS.	Scegli la regione principale, che hai specificato nella prima epopea.	DevOps

Fase di costruzione: crea e configura il progetto

Operazione	Description	Competenze richieste
Creazione del progetto	Scegli Crea progetto e inserisci un nome per il progetto.	DevOps
Specificate l'immagine dell'ambiente.	Per questa dimostrazione del pattern, utilizzate l'immagin e CodeBuild gestita predefini ta. Hai anche la possibilità di	DevOps

Operazione	Description	Competenze richieste
	utilizzare un'immagine Docker personalizzata, se ne hai una.	
Specificate il sistema operativo.	Scegli Amazon Linux 2 o Ubuntu. **Note** Amazon Linux 2 sta per terminare il supporto. Per ulteriori informazioni, consulta Amazon Linux 2 FAQs.	DevOps
Specificare il ruolo del servizio.	Scegli il ruolo per cui hai creato CodeBuild prima di iniziare a creare il CodePipel ine lavoro. (Vedi la sezione Prerequisiti).	DevOps
Imposta opzioni aggiuntive.	Per Timeout e Queued timeout, mantieni i valori predefiniti. Per il certifica to, mantieni l'impostazione predefinita a meno che tu non abbia un certificato personali zzato da utilizzare.	DevOps
Crea le variabili di ambiente.	Per ogni regione AWS in cui desideri effettuare la distribuz ione, crea variabili di ambiente fornendo il nome del bucket S3 e il nome della regione (ad esempio, us-east-1).	DevOps

Operazione	Description	Competenze richieste
Fornisci il nome del file buildspec, se non è buildspec .yml.	Mantieni vuoto questo campo se il nome del file è quello predefinito,. `buildspec .yaml` Se hai rinominato il file buildspec, inserisci il nome qui. Assicurati che corrispon da al nome del file che si trova nel repository. CodeCommit	DevOps
Specificare la registrazione.	Per visualizzare i log di Amazon CloudWatch Events, mantieni l'impostazione predefinita. Oppure puoi definire nomi di gruppi o logger specifici.	DevOps

Salta la fase di distribuzione

Operazione	Description	Competenze richieste
Salta la fase di implement azione e completa la creazione della pipeline.	Quando si configura la pipeline, CodePipeline consente di creare solo una fase nella fase di distribuz ione. Per eseguire la distribuz ione in più regioni AWS, salta questa fase. Dopo aver creato la pipeline, puoi aggiungere più fasi della fase di distribuz ione.	DevOps

Fase di distribuzione: configura la pipeline per la distribuzione nella prima regione

Operazione	Description	Competenze richieste
Aggiungi una fase alla fase di implementazione.	Modifica la pipeline e scegli Aggiungi fase nella fase di distribuzione. Questa prima fase è per la regione principal e.	DevOps
Fornisci un nome di azione per la fase.	Inserisci un nome univoco che rifletta la prima fase (principa le) e la regione. <region>Ad esempio, inserisci primary_ _deploy.	DevOps
Specificare il fornitore dell'azio ne.	Per il provider Action, scegli AWS CloudFormation.	DevOps
Configura la regione per la prima fase.	Scegli la prima regione (principale), la stessa regione in cui CodePipeline CodeBuild sono configurate. Questa è la regione principale in cui desideri distribuire lo stack.	DevOps
Specificate l'artefatto di input.	Scegli BuildArtifact. Questo è l'output della fase di costruzio ne.	DevOps
Specificate l'azione da intraprendere.	Per la modalità Azione, scegli Crea o aggiorna uno stack.	DevOps
Inserisci un nome per lo CloudFormation stack.		DevOps
Specificate il modello per la prima regione.	Seleziona il nome del pacchetto specifico della regione che è stato impacchet	DevOps

Operazione	Description	Competenze richieste
	tato da CodeBuild e scaricato nel bucket S3 per la prima regione (primaria).	
Specificate le funzionalità.	Le funzionalità sono necessari e se il modello di stack include risorse IAM o se si crea uno stack direttamente da un modello che contiene macro. Per questo modello, usa CAPABILITY_IAM, CAPABILITY_NAMED_IAM, CAPABILITY_AUTO_EXPAND.	DevOps

Fase di distribuzione: configura la pipeline per la distribuzione nella seconda regione

Operazione	Description	Competenze richieste
Aggiungi la seconda fase alla fase di distribuzione.	Per aggiungere una fase per la seconda regione, modifica la pipeline e scegli Aggiungi fase nella fase di distribuzione. Importante: il processo di creazione della seconda regione è lo stesso di quello della prima regione, ad eccezione dei seguenti valori.	DevOps
Fornisci un nome di azione per la seconda fase.	Inserisci un nome univoco che rifletta la seconda fase e la seconda regione.	DevOps
Configura la regione per la seconda fase.	Scegli la seconda regione in cui desideri distribuire lo stack.	DevOps

Operazione	Description	Competenze richieste
Specificate il modello per la seconda regione.	Seleziona il nome del pacchetto specifico della regione che è stato impacchet tato da CodeBuild e scaricato nel bucket S3 per la seconda regione.	DevOps

Fase di distribuzione: configura la pipeline per la distribuzione nella terza regione

Operazione	Description	Competenze richieste
Aggiungi la terza fase alla fase di distribuzione.	Per aggiungere una fase per la terza regione, modifica la pipeline e scegli Aggiungi fase nella fase di distribuz ione. Importante: il processo di creazione della seconda regione è lo stesso delle due regioni precedenti, ad eccezione dei seguenti valori.	DevOps
Fornisci un nome di azione per la terza fase.	Inserisci un nome univoco che rifletta la terza fase e la terza regione.	DevOps
Configura la regione per la terza fase.	Scegli la terza regione in cui desideri distribuire lo stack.	DevOps
Specificate il modello per la terza regione.	Seleziona il nome del pacchetto specifico della regione che è stato impacchet tato da CodeBuild e scaricato nel bucket S3 per la terza regione.	DevOps

Pulisci la distribuzione

Operazione	Description	Competenze richieste
Elimina le risorse AWS.	Per ripulire la distribuzione, elimina gli CloudFormation stack in ogni regione. Quindi elimina CodeCommit CodeBuild le CodePipeline risorse e dalla regione principale.	DevOps

Risorse correlate

- Che cos'è AWS CodePipeline?
- Modello di applicazione serverless AWS
- AWS CloudFormation
- Riferimento alla struttura CloudFormation dell'architettura AWS per AWS CodePipeline

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Distribuisci carichi di lavoro dalle DevOps pipeline di Azure ai cluster Amazon EKS privati

Mahendra Revanasiddappa, Amazon Web Services

Riepilogo

Questo modello dimostra come implementare l'integrazione continua e la distribuzione continua (CI/CD) dalle DevOps pipeline di Azure ai cluster privati di Amazon Elastic Kubernetes Service (Amazon EKS). Risolve una sfida critica affrontata dalle organizzazioni che stanno migliorando il proprio livello di sicurezza passando a endpoint server API privati per i propri cluster Amazon EKS.

Un endpoint pubblico espone il server dell'API Kubernetes direttamente a Internet, creando una superficie di attacco più ampia che i malintenzionati potrebbero potenzialmente prendere di mira. Passando a un endpoint privato, l'accesso al piano di controllo del cluster è limitato all'interno del cloud privato virtuale (VPC) del cliente.

Sebbene la transizione di un cluster Amazon EKS a un endpoint API privato migliori in modo significativo la sicurezza, introduce sfide di connettività per piattaforme esterne CI/CD come Azure. DevOps L'endpoint privato è accessibile solo dall'interno del VPC o dalle reti peer del cluster. Pertanto, DevOps gli agenti Azure standard ospitati da Microsoft, che operano all'esterno della rete AWS privata, non possono raggiungere direttamente il server dell'API Kubernetes. Ciò interrompe i flussi di lavoro di distribuzione tipici che si basano su strumenti come kubectl o Helm in esecuzione su questi agenti perché non riescono a stabilire una connessione al cluster.

Per ovviare a questo problema, questo modello mostra un approccio efficiente utilizzando DevOps agenti di Azure ospitati autonomamente all'interno di cluster Amazon EKS privati. Questa soluzione offre un'ottimizzazione dei costi, un'efficienza operativa e una scalabilità superiori, preservando al contempo i requisiti di sicurezza. Questo approccio è particolarmente vantaggioso per le aziende che cercano di semplificare i DevOps processi multi-cloud senza compromettere le prestazioni o la sicurezza.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- AWS Command Line Interface (AWS CLI) installata la versione 2.13.17 o successiva.

- kubectl versione 1.25.1 o successiva, installata.
- Creazione di un cluster Amazon EKS privato versione 1.24 o successiva, con autorizzazioni per creare namespace, segreti e distribuzioni.
- Nodi di lavoro in un cluster Amazon EKS con connettività in uscita a Internet in modo che l' DevOps agente di Azure in esecuzione su di essi possa connettersi al pool di agenti di Azure DevOps .
- GitHub account creato.
- È stato creato un DevOps progetto Azure con accesso alla configurazione delle connessioni di servizio, che sono connessioni autenticate tra Azure Pipelines e servizi esterni o remoti.
- La AWS Toolkit per Azure DevOps versione 1.15 o successiva installata per il DevOps progetto Azure descritto nel punto precedente. Per istruzioni di installazione, consulta AWS Toolkit per Azure DevOpsVisual Studio Marketplace.

Limitazioni

- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità regionale, vedi AWS Servizi per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Architecture

Questo modello crea quanto segue:

- Repository Amazon ECR - Il repository Amazon Elastic Container Registry (Amazon ECR) archivia l'immagine Docker con l'agente DevOps Azure e l'app di esempio distribuita.
- Pool di DevOps agenti di Azure: un pool di agenti con DevOps hosting autonomo di Azure registra l'agente in esecuzione sul cluster privato Amazon EKS.
- Ruolo IAM: un ruolo AWS Identity and Access Management (IAM) per la connessione al servizio Azure per fornire l'accesso richiesto all'agente in esecuzione su un cluster Amazon EKS privato.
- Connessione DevOps al servizio Azure: una connessione di servizio in un DevOps account Azure per usare il ruolo IAM che fornisce l'accesso richiesto per l'accesso ai job della pipeline. Servizi AWS

Il diagramma seguente mostra l'architettura di distribuzione di un DevOps agente di Azure con hosting autonomo su un cluster Amazon EKS privato e la distribuzione di un'applicazione di esempio sullo stesso cluster.

Il diagramma mostra il flusso di lavoro seguente:

1. Implementa un DevOps agente di Azure ospitato autonomamente come distribuzione all'interno di un cluster Amazon EKS.
2. Un DevOps agente di Azure si connette al pool di agenti su un DevOps account Azure usando un token di accesso personale (PAT) per l'autenticazione.
3. Azure Pipelines configura una pipeline da distribuire usando il codice di un repository. GitHub
4. La pipeline viene eseguita sull'agente dal pool di agenti configurato nella configurazione della pipeline. L' DevOps agente di Azure ottiene le informazioni sul lavoro della pipeline effettuando costantemente il polling sull'account Azure. DevOps
5. L' DevOps agente di Azure crea un'immagine Docker come parte del processo della pipeline e la invia al repository Amazon ECR.
6. L' DevOps agente di Azure distribuisce l'applicazione di esempio su un cluster Amazon EKS privato in uno spazio dei nomi chiamato. `webapp`

Tools (Strumenti)

Strumenti

- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire AWS Kubernetes senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.

Altri strumenti

- Docker è un insieme di prodotti Platform as a Service (PaaS) che utilizzano la virtualizzazione a livello di sistema operativo per fornire software in container.
- kubectl è un'interfaccia a riga di comando che consente di eseguire comandi sui cluster Kubernetes.

Archivio di codice

- Il codice per questo pattern è disponibile nel repository GitHub deploy-kubernetes-resources-to-amazon-eks-using-azure -devops.

Best practice

- Per Amazon EKS, consulta la Amazon EKS Best Practice Guide.
- Segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.

Epiche

Crea una connessione di servizio

Operazione	Description	Competenze richieste
Trova il GUID dell' DevOps organizzazione di Azure.	Accedi al tuo DevOps account Azure, quindi usa il seguente URL per trovare il GUID dell'organizzazione: `https://dev.azure.com/{DevOps_Org_ID}/_apis/projectCollections?api-version=6.0` Nell'URL, sostituisci `{DevOps_org_ID}` con il tuo ID dell'organizzazione Azure. DevOps	AWS DevOps
Configura un IdP in. Account AWS	Per configurare un provider di identità (IdP) Account AWS per una connessione al servizio Azure, usa i seguenti passaggi: 1. Accedi a e apri Console di gestione AWS la console	AWS DevOps

Operazione	Description	Competenze richieste
	IAM all'indirizzo. https://console.aws.amazon.com/iam/ 2. Nel riquadro di sinistra, scegli Provider di identità. 3. Scegliere Aggiungi provider. 4. Scegli OpenID Connect come tipo di provider. 5. Per l'URL del provider, inserisci l'URL dell' DevOps emittente di Azure. Ogni tenant di Azure ne DevOps ha uno univoco `OrganizationGUID` che in genere usa il seguente formato: `https://vstoken.dev.azure.com/{OrganizationGUID}` Sostituisci `{OrganizationGUID}` con l'ID dell'organizzazione di Azure. DevOps 6. Per Audience, inserisci api: //Azure Exchange. ADToken Si tratta di un valore fisso per Azure. DevOps 7. Scegliere Aggiungi provider. 8. Prendi nota dell'ARN del provider appena creato da utilizzare nell'attività successiva.	

Operazione	Description	Competenze richieste
	Per altri dettagli, vedi Come eseguire la federazione AWS da Azure usando DevOps OpenID Connect.	

Operazione	Description	Competenze richieste
Crea una policy IAM in Account AWS.	Per creare una policy IAM che fornisca le autorizza zioni richieste al ruolo IAM usato dalla DevOps pipeline di Azure, usa i seguenti passaggi: 1. Nella console IAM, nel riquadro a sinistra, scegli Policies. 2. Scegli Crea policy. 3. Per Specificare le autorizza zioni, nell'editor delle politiche, seleziona JSON. Sostituisci la policy JSON predefinita con la seguente JSON: `{ "Version": "2012-10-17", "Statement": [ { "Sid": "Statement1", "Effect": "Allow", "Action": [ "ecr:*", "eks:DescribeClust er", "eks:ListClusters" ],`	AWS DevOps

Operazione	Description	Competenze richieste

	<pre>"Resource
	": "*"
	}
	]
	}</pre>
	4. Scegli Next (Successivo).
	5. Per il nome della policy, inserisci un nome per la policy IAM. Questo modello utilizza il nome ADO-Policy.
	6. Scegli Crea policy.	

Operazione	Description	Competenze richieste
Crea un ruolo IAM in Account AWS.	Per configurare un ruolo IAM nella Account AWS connessione al servizio Azure, usa i seguenti passaggi: 1. Nella console IAM, nel riquadro a sinistra, scegli Ruoli. 2. Scegli Crea ruolo. 3. Per il tipo di entità affidabile, seleziona Web Identity. 4. Seleziona l'IdP corretto dall'elenco a discesa. Il nome IdP inizia con. `vstoken.dev.azure.com/{OrganizationGUID}` 5. Nell'elenco a discesa Audience, seleziona api: ADToken //Azure Exchange. 6. Per limitare questo ruolo a una sola connessione di servizio, aggiungi una condizione. In Condizione, seleziona Aggiungi condizione e per Chiave, seleziona vstoken.dev.azure.com/ {organizationGUID} :sub. StringEqualsIn Condizione, seleziona. Per Valore, usa il seguente formato:`sc://{OrganizationName}/`	AWS DevOps

Operazione	Description	Competenze richieste
	`{ProjectName}/{ServiceConnectionName}` . Per `ServiceConnectionName` , usa aws-sc. Creerai questa connessione al servizio nella prossima attività. 7. Scegli Next (Successivo). 8. Per Aggiungi autorizzazioni, seleziona ADO-Policy, che è la politica che hai creato in un'attività precedente. 9. Scegli Avanti e, per Nome del ruolo, inserisci ado-role. Per Select Trusted Entities, utilizza la seguente politica di fiducia: `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "Federated": "arn:aws:iam::{account_id}:oidc-provider/vstoken.dev.azure.com/{OrganizationGUID}" }, "Action": "sts:AssumeRoleWithWebIdentity", "Condition": {`	

Operazione	Description	Competenze richieste

	<pre>"StringEquals": {
	"vstoken.dev.azure.com/{OrganizationGUID}:aud": "api://AzureADTokenExchange",
	"vstoken.dev.azure.com/{OrganizationGUID}:sub": "sc://{OrganizationName}/{ProjectName}/{ServiceConnectionName}"
	}
	}
	}
	]
	}</pre> Nella politica, fornisci le tue informazioni per i seguenti segnaposto:
	• `{account_id}` - ID Account AWS
	• `{OrganizationGUID}` - GUID DevOps dell'organizzazione di Azure
	• `{OrganizationName}` - Nome dell'organizzazione di Azure DevOps
	• `{ProjectName}` - Nome del progetto Azure DevOps
	• `{ServiceConnectionName}` - Nome della connessione al DevOps	

Operazione	Description	Competenze richieste
	servizio Azure. Usa aws-sc. Creerai questa connessio ne al servizio nella prossima attività.	

Operazione	Description	Competenze richieste
Crea una connessione di servizio nell' DevOps account Azure.	Per configurare una connessione al servizio Azure, usa i seguenti passaggi: 1. Nel tuo progetto Azure, seleziona Impostazioni DevOps del progetto, Connessioni ai servizi. 2. Scegli Nuova connessione al servizio, seleziona il tipo di connessione al servizio come`aws`, quindi seleziona Avanti. 3. In Role to Assume, inserisci l'arn per il ruolo IAM ado-role. Hai creato un ruolo IAM nel task precedente. Crea un ruolo IAM in. Account AWS 4. Seleziona la casella di controllo Usa OIDC. 5. Per il nome della connessione al servizio, inserisci aws-sc nelle proprietà dell'attività. 6. Scegli Save (Salva). Per ulteriori dettagli, consulta Creare una connessione di servizio nella documentazione Microsoft.	AWS DevOps

Operazione	Description	Competenze richieste

Aggiungi il ruolo IAM al file di configurazione di Amazon EKS.	Il ruolo IAM deve disporre delle autorizzazioni necessari e per eseguire le operazioni richieste sul cluster Amazon EKS. Poiché si tratta di un ruolo di pipeline, il ruolo IAM deve essere in grado di gestire quasi tutti i tipi di risorse sul cluster. Pertanto, l'autoriz zazione `system:masters` di gruppo è appropriata per questo ruolo.
	
	Per aggiungere la configura zione richiesta all'`aws-auth ConfigMap` interno di Kubernetes, usa il codice seguente:
	
	`- groups:`
	`  - system:masters`
	`  rolearn: arn:aws:i am::{account_id}:r ole/ADO-role`
	`  username: ADO-role`
	
	Sostituiscilo `{account_id}` con il tuo ID. Account AWS
	
	Per ulteriori informazioni, consulta Come funziona Amazon EKS con IAM nella documentazione di Amazon EKS.
		AWS DevOps

Crea un pool di agenti

Operazione	Description	Competenze richieste
Crea un pool di agenti ospitati autonomamente.	Per configurare un pool di agenti ospitati autonomamente nell' DevOps account Azure, usa i seguenti passaggi: 1. Accedi all'organizzazione del tuo account Azure. DevOps 2. Scegli Azure Organization. DevOps 3. Scegli il tuo progetto Azure. DevOps 4. Selezionare Project settings (Impostazioni del progetto). 5. Scegli Agent pool. 6. Scegli Aggiungi pool. 7. Seleziona Self-hosted. 8. Per Nome, inserisci eks-agent. 9. Seleziona la casella di controllo Concedi l'autoriz zazione di accesso a tutte le pipeline. 10. Scegli Create (Crea). Per ulteriori dettagli, consulta Creare e gestire pool di agenti nella documentazione Microsoft.	

Crea l'immagine DevOps dell'agente di Azure e inviala ad Amazon ECR

Operazione	Description	Competenze richieste
Crea un repository Amazon ECR.	Le immagini Docker utilizzate per distribuire l' DevOps agente Azure e l'applicazione di esempio (`webapp`) sul cluster Amazon EKS privato devono essere archiviate in un repository Amazon ECR. Per creare un repository Amazon ECR, utilizza i seguenti passaggi: 1. Apri la console Amazon ECR nei https://console.aws.amazon.com/ecr/repository. 2. Dalla barra di navigazione, scegli in cui Regione AWScreare il tuo repository. 3. Nella pagina Repository, scegli Archivi privati, quindi scegli Crea repository. 4. Per il nome del repository, inserisci webapp. Affinché l'applicazione di esempio in questo modello funzioni, il nome del repository Amazon ECR deve utilizzare webapp. Se utilizzi un nome diverso per il repository, consulta Risoluzione dei problemi.	AWS DevOps

Operazione	Description	Competenze richieste
	Per ulteriori dettagli, consulta Creazione di un repository privato Amazon ECR per archiviare immagini nella documentazione di Amazon ECR.	

Operazione	Description	Competenze richieste

Crea un Dockerfile per creare l'agente Azure DevOps .	Crea un Dockerfile per creare l'immagine Docker su cui è installato l'agente Azure. DevOps Archivia il seguente contenuto in un file denominato: `Dockerfile`
	
	`FROM ubuntu:22.04`
	`ENV TARGETARCH="linux-x64"`
	`RUN apt update && apt upgrade -y && apt install -y curl git jq libicu70 unzip wget`
	
	`RUN curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"`
	`RUN unzip awscliv2.zip`
	`RUN ./aws/install`
	`RUN rm -rf aws awscliv2.zip`
	
	`RUN curl -sSL https://get.docker.com/ | sh`
	
	`RUN curl -sL https://aka.ms/InstallAzureCLIDeb | bash`
	`RUN mkdir -p azp`
	`WORKDIR /azp/`
	
	`COPY ./start.sh ./`
	`RUN chmod +x ./start.sh`
	
	`RUN useradd -m -d /home/agent agent`
		AWS DevOps

Operazione	Description	Competenze richieste

	`RUN chown -R agent:agent /azp /home/agent`
	`RUN groupadd -f docker`
	`RUN usermod -aG docker agent`
	`USER agent`
	
	`ENTRYPOINT [ "./start.sh" ]`	

Crea script per l' DevOps agente di Azure.	Per creare lo `start.sh` script, usa i seguenti passaggi: 1. Vai alla procedura Crea e crea il Dockerfile nella documentazione di Microsoft e scorri fino al passaggio 5. Salva il seguente contenuto in `~/azp-agent-in-docker/start.sh` , assicurandoti di utilizzare terminazioni di riga in stile Unix (LF). 2. Copia il contenuto dello script e salvalo in un file denominato `start.sh` nella stessa directory del Dockerfile.	AWS DevOps

Operazione	Description	Competenze richieste

Crea un'immagine Docker con l'agente Azure. DevOps	Per creare un'immagine Docker per installare l' DevOps agente di Azure, usa il Dockerfile che hai creato in precedenza per creare l'immagine. Nella stessa directory in cui è archiviato il Dockerfile, esegui i seguenti comandi:
	`aws ecr get-login-password --region region | docker login --username AWS --password-stdin aws_account_id.dkr.ecr.region.amazonaws.com`
	
	`docker build --platform linux/amd64 -t ado-agent:latest .`
	
	`docker tag ado-agent:latest aws_account_id.dkr.ecr.region.amazonaws.com/webapp:latest`
	
	`docker push aws_account_id.dkr.ecr.region.amazonaws.com/webapp:latest`
	
	Sostituisci `aws_account_id` e `region` con il tuo Account AWS ID e. Regione AWS
		AWS DevOps

Distribuisci l' DevOps agente di Azure in un cluster Amazon EKS privato

Operazione	Description	Competenze richieste
Genera un token di accesso personale di Azure.	L'agente in esecuzione sul cluster privato Amazon EKS richiede un token di accesso personale (PAT) per potersi autenticare con l'account Azure DevOps . Per generare un PAT, usa i seguenti passaggi: 1. Accedi con l'account utente che intendi usare nella tua DevOps organizza zione Azure ()`https://dev.azure.com/{Your_Organization}` . 1. Dalla tua home page, apri le impostazioni utente, quindi seleziona Token di accesso personali. 2. Scegli Nuovo token. 3. Inserisci il nome per il token. 4. Scegli Mostra tutti gli ambiti. 5. Per Agent Pools, seleziona la casella di controllo Leggi e gestisci. 6. Scegli Create (Crea). 7. Per creare un segreto sul cluster Amazon EKS	AWS DevOps

Operazione	Description	Competenze richieste

	privato, utilizza la seguente configurazione:
	
	`apiVersion: v1`
	`kind: Secret`
	`metadata:`
	`  name: azdevops-pat`
	`  namespace: default`
	`type: Opaque`
	`stringData:`
	`  AZP_TOKEN: <PAT Token>`
	
	1. Memorizza la configurazione in un file denominato `ado-secret.yaml` . Sostituiscila `<PAT Token>` con il token di accesso personale che hai appena creato. Per creare il segreto, esegui il seguente comando:
	
	`kubectl create -f ado-secret.yaml`
	
	Per ulteriori dettagli, consulta Registrare un agente utilizzando un token di accesso personale (PAT) nella documentazione Microsoft.	

Operazione	Description	Competenze richieste

Utilizza il file manifest di Kubernetes per la distribuzione degli agenti.	Per distribuire l' DevOps agente di Azure sul cluster privato Amazon EKS, copia il seguente file manifest e archivia il file come: `agent-deployment.yaml`
	
	`apiVersion: apps/v1`
	`kind: Deployment`
	`metadata:`
	`  name: azure-pipelines-agent-eks`
	`  labels:`
	`  app: azure-pipelines-agent`
	`spec:`
	`  replicas: 1`
	`  selector:`
	`  matchLabels:`
	`  app: azure-pipelines-agent`
	`  template:`
	`  metadata:`
	`  labels:`
	`  app: azure-pipelines-agent`
	`  spec:`
	`  containers:`
	`  - name: docker`
	`  image: docker:dind`
	`  securityContext:`
	`  privileged: true`
	`  volumeMounts:`
	`  - name: shared-workspace`
	`  mountPath: /workspace`
		AWS DevOps

Operazione	Description	Competenze richieste

	`- name: dind-storage`
	`mountPath: /var/lib/docker`
	`env:`
	`- name: DOCKER_TLS_CERTDIR`
	`value: ""`
	`- name: azure-pipelines-agent`
	`image: aws_account_id.dkr.ecr.region.amazonaws.com/webapp:latest`
	`env:`
	`- name: AZP_URL`
	`value: "<Azure account URL>"`
	`- name: AZP_POOL`
	`value: "eks-agent"`
	`- name: AZP_TOKEN`
	`valueFrom:`
	`secretKeyRef:`
	`name: azdevops-pat`
	`key: AZP_TOKEN`
	`- name: AZP_AGENT_NAME`
	`valueFrom:`
	`fieldRef:`
	`fieldPath: metadata.name`
	`- name: DOCKER_HOST`
	`value: tcp://localhost:2375`
	`volumeMounts:`	

Operazione	Description	Competenze richieste

	<pre>- mountPath: /workspace
	name: shared-workspace
	volumes:
	- name: dind-storage
	emptyDir: {}
	- name: shared-workspace
	emptyDir: {}</pre>
	Sostituisci `aws_account_id` e `<Azure account URL>` con il tuo Account AWS ID e l'URL dell'account Azure. DevOps	

Implementa l'agente sul cluster privato Amazon EKS.	Per distribuire l'agente Azure Devops sul cluster privato Amazon EKS, usa il seguente comando: <pre>kubectl create -f agent-deployment.tf</pre>	AWS DevOps

Operazione	Description	Competenze richieste

Verifica che l'agente sia in esecuzione.	Per verificare che l' DevOps agente di Azure sia in esecuzione, usa il seguente comando:
	`kubectl get deploy azure-pipelines-agent-eks`
	L'output previsto dovrebbe essere simile al seguente:
	`NAME READY UP-TO-DATE AVAILABLE AGE azure-pipelines-agent-eks 1/1 1 1 58s`
	Assicurati che la `READY` colonna sia visualizzata`1/1`.
		AWS DevOps

Operazione	Description	Competenze richieste
Verifica che l'agente sia registrato nel pool di DevOps agenti di Azure.	Per verificare che l'agente sia distribuito nel cluster privato Amazon EKS e sia registrato nel pool di agenti`eks-agent` , utilizza i seguenti passaggi: 1. Accedi alla tua DevOps organizzazione Azure ()`https://dev.azure.com/{Your_Organization}` . 2. Selezionare Project settings (Impostazioni del progetto). 3. Scegli Agent pool. 4. Seleziona il pool eks-agent , quindi controlla la scheda Agenti. Dovresti vedere un agente elencato con lo stato Online e il nome dell'agente dovrebbe iniziare con azure-pipelines-agent-eks -*.	AWS DevOps

Distribuisci un'applicazione di esempio

Operazione	Description	Competenze richieste
Inserisci il repository di applicazioni di esempio nel tuo GitHub account.	Effettua il fork del seguente repository AWS di esempi sul tuo account: GitHub https://github.com/aws-samples/deploy-kubernetes-resou	AWS DevOps

Operazione	Description	Competenze richieste
	rces-to- -devops amazon-eks-using-azure	

Operazione	Description	Competenze richieste
Creare una pipeline.	Per creare una pipeline nel tuo DevOps account Azure, usa i seguenti passaggi: 1. Accedi con l'account utente che intendi usare nella tua DevOps organizza zione Azure (). `https://dev.azure.com/{Your_Organization}` 2. Accedi alla console del progetto e delle pipeline. 3. Scegli New Pipeline. 4. Per Dov'è il tuo codice, scegliGitHub. 5. Fornisci le credenziali necessarie affinché la pipeline si connetta al tuo account GitHub 6. Seleziona il repositor y deploy-kubernetes-resources-to - -devops. amazon-eks-using-azure 7. Per Configura la pipeline, seleziona il file YAML di Azure Pipelines esistente. 8. Per Seleziona un file YAML esistente, seleziona main per Branch e azure_pip elines.yaml per percorso. 9. Scegli Continua. 10. Per Review your pipeline YAML, sostituisci i valori	AWS DevOps

Operazione	Description	Competenze richieste

	dei parametri di input per e con le tue informazioni: `awsRegion` `awsEKSClusterName`
	
	`pool:`
	`  name: eks-agent`
	`#pool: self-hosted # If you are running self-hosted Azure DevOps Agents`
	
	`stages:`
	`# Refering the pipeline template, input parameter that are not specified will be added with defaults`
	`- template: ./pipeline_templates/main_template.yaml`
	`  parameters:`
	`  serviceConnectionName: aws-sc`
	`  awsRegion: <your region>`
	`  awsEKSClusterName: <name of your EKS cluster>`
	`  projectName: webapp`
	
	1. Scegli RUN.	

Operazione	Description	Competenze richieste
Verifica che l'applicazione di esempio sia stata distribuita.	Una volta completata la pipeline, verifica la corretta implementazione dell'appl icazione di esempio controlla ndo sia il repository Amazon ECR che il cluster Amazon EKS. Per verificare gli artefatti nel repository Amazon ECR, utilizza i seguenti passaggi: 1. Accedi al repository `webapp` Amazon ECR. 2. Conferma la presenza dei seguenti nuovi artefatti: • Immagine Docker: `<date>.<build_numb er>-image` • Diagramma di timone — `<date>.<build_numb er>-helm` Ad esempio `20250501.1-image` e `20250501.1-helm`. Per verificare la distribuz ione sul cluster Amazon EKS privato nello spazio dei nomi`webapp`, usa il seguente comando:	AWS DevOps

Operazione	Description	Competenze richieste

	`kubectl get deploy -n webapp`
	
	L'output previsto è il seguente:
	
	`NAME READY UP-TO-DATE AVAILABLE`
	`webapp 1/1 1 1`
	
	Nota: se questa è la prima esecuzione della pipeline, potrebbe essere necessario autorizzare la connessione al servizio e il pool di agenti. Cerca le richieste di autorizza zione nell'interfaccia della DevOps pipeline di Azure e approvale per procedere.	

Risoluzione dei problemi

Problema	Soluzione
La pipeline fallisce quando il nome del repositor y Amazon ECR non corrisponde `webapp`	L'applicazione di esempio si aspetta che il nome del repository Amazon ECR corrispon da al `projectName: webapp` parametro in. `azure_pipeline.yml` Per risolvere questo problema, rinomina il tuo repository Amazon ECR o aggiorna `webapp` quanto segue:

Problema	Soluzione
	• Rinomina la `webapp` directory nel tuo repository biforcato in modo che corrisponda al nome del tuo GitHub repository Amazon ECR. • Aggiorna il `projectName` parametro in `azure_pipeline.yml` modo che corrisponda al nome del tuo repository Amazon ECR.
Errore: cluster Kubernetes irraggiungibile: il server ha richiesto al client di fornire le credenziali	Se riscontri questo errore nella fase «Pull and Deploy Helm Chart» nella pipeline di Azure, la causa principale in genere deriva da una configurazione errata del ruolo IAM nel tuo cluster Amazon EKS. `aws-auth ConfigMap` Per risolvere questo problema, controlla quanto segue: • Verifica la tua `aws-auth ConfigMap` configurazione. • Controlla le impostazioni di autenticazione del tuo cluster Amazon EKS: apri la console Amazon EKS, i dettagli del cluster, la configurazione di Access. Assicurati che la modalità di autenticazione sia impostata sull'API EKS e ConfigMap (non solo sull'API EKS).

Risorse correlate

AWS Blog

- Come eseguire la federazione AWS da Azure usando DevOps OpenID Connect

Servizi AWS documentazione

- Amazon ECR
- Amazon EKS

Documentazione Microsoft

- Che cos'è Azure DevOps?
- Che cos'è Azure Pipelines?

Esegui le query SQL di Amazon Redshift utilizzando Terraform

Sylvia Qi e Aditya Ambati, Amazon Web Services

Riepilogo

L'uso dell'infrastruttura come codice (IaC) per la distribuzione e la gestione di Amazon Redshift è una pratica prevalente all'interno. DevOps IaC facilita l'implementazione e la configurazione di varie risorse Amazon Redshift, come cluster, snapshot e gruppi di parametri. Tuttavia, IaC non si estende alla gestione di risorse di database come tabelle, schemi, viste e stored procedure. Questi elementi del database sono gestiti tramite query SQL e non sono supportati direttamente dagli strumenti IaC. Sebbene esistano soluzioni e strumenti per la gestione di queste risorse, potresti preferire non introdurre strumenti aggiuntivi nel tuo stack tecnologico.

Questo modello delinea una metodologia che utilizza Terraform per distribuire le risorse del database Amazon Redshift, tra cui tabelle, schemi, viste e stored procedure. Il modello distingue tra due tipi di query SQL:

- Query non ripetibili: queste query vengono eseguite una sola volta durante la distribuzione iniziale di Amazon Redshift per stabilire i componenti essenziali del database.
- Query ripetibili: queste query sono immutabili e possono essere rieseguite senza influire sul database. La soluzione utilizza Terraform per monitorare le modifiche nelle query ripetibili e applicarle di conseguenza.

Per maggiori dettagli, consulta la procedura dettagliata della soluzione in Informazioni aggiuntive.

Prerequisiti e limitazioni

Prerequisiti

È necessario disporre di un dispositivo attivo Account AWS e installare quanto segue sul computer di distribuzione:

- AWS Command Line Interface (AWS CLI)
- Un AWS CLI profilo configurato con le autorizzazioni di Amazon Redshift read/write
- Terraform versione 1.6.2 o successiva
- Python 3

- Boto3

Limitazioni

- Questa soluzione supporta un singolo database Amazon Redshift perché Terraform consente la creazione di un solo database durante la creazione del cluster.
- Questo modello non include test per convalidare le modifiche alle query ripetibili prima di applicarle. Si consiglia di incorporare tali test per una maggiore affidabilità.
- Per illustrare la soluzione, questo modello fornisce un file di esempio che utilizza un `redshift.tf` file di stato Terraform locale. Tuttavia, per gli ambienti di produzione, si consiglia vivamente di utilizzare un file di stato remoto con un meccanismo di blocco per una maggiore stabilità e collaborazione.
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Versioni del prodotto

Questa soluzione è stata sviluppata e testata sulla patch 179 di Amazon Redshift.

Repository di codice

Il codice per questo modello è disponibile nel repository GitHub amazon-redshift-sql-deploy-terraform.

Architecture

Il diagramma seguente illustra come Terraform gestisce le risorse del database Amazon Redshift gestendo query SQL sia non ripetibili che ripetibili.

Il diagramma mostra i seguenti passaggi:

1. Terraform applica query SQL non ripetibili durante la distribuzione iniziale del cluster Amazon Redshift.
2. Lo sviluppatore apporta modifiche alle query SQL ripetibili.
3. Terraform monitora le modifiche nelle query SQL ripetibili.
4. Terraform applica query SQL ripetibili al database Amazon Redshift.

La soluzione fornita da questo modello è costruita sulla base del modulo Terraform per Amazon Redshift. Il modulo Terraform fornisce un cluster e un database Amazon Redshift. Per migliorare il modulo, abbiamo utilizzato `terraform_data` risorse, che richiamano uno script Python personalizzato per eseguire query SQL utilizzando l'operazione API Amazon Redshift. ExecuteStatement Di conseguenza, il modulo può eseguire le seguenti operazioni:

- Distribuisci un numero qualsiasi di risorse del database utilizzando le query SQL dopo il provisioning del database.
- Monitora continuamente le modifiche nelle query SQL ripetibili e applica tali modifiche utilizzando Terraform.

Per maggiori dettagli, consulta la procedura dettagliata della soluzione in Informazioni aggiuntive.

Tools (Strumenti)

Servizi AWS

- Amazon Redshift è un servizio di data warehouse su scala petabyte completamente gestito in. Cloud AWS

Altri strumenti

- Terraform è uno strumento di infrastruttura come codice (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.
- Python è un linguaggio di programmazione generico utilizzato in questo modello per eseguire query SQL.

Best practice

- Best practice di Amazon Redshift
- Utilizzo dell'API Amazon Redshift Data per interagire con i cluster Amazon Redshift

Epiche

Implementa la soluzione utilizzando Terraform

Operazione	Description	Competenze richieste
Clona il repository.	Per clonare il repository Git contenente il codice Terraform per il provisioning di un cluster Amazon Redshift, usa il seguente comando. `git clone https://github.com/aws-samples/amazon-redshift-sql-deploy-terraform.git`	DevOps ingegnere

Aggiorna le variabili Terraform.	Per personalizzare la distribuzione del cluster Amazon Redshift in base ai tuoi requisiti specifici, aggiorna i seguenti parametri nel `terraform.tfvars` file.
	`region = "<AWS_REGION>"`
	`cluster_identifier = "<REDSHIFT_CLUSTER_IDENTIFIER>"`
	`node_type = "<REDSHIFT_NODE_TYPE>"`
	`number_of_nodes = "<REDSHIFT_NODE_COUNT>"`
	`database_name = "<REDSHIFT_DB_NAME>"`
		DevOps ingegnere

Operazione	Description	Competenze richieste

	`subnet_ids = "<REDSHIFT_SUBNET_IDS>"`
	`vpc_security_group_ids = "<REDSHIFT_SECURITY_GROUP_IDS>"`
	`run_nonrepeatable_queries = true`
	`run_repeatable_queries = true`
	`sql_path_bootstrap = "<BOOTSTRAP_SQLS_PATH>"`
	`sql_path_nonrepeatable = "<NON-REPEATABLE_SQLS_PATH>"`
	`sql_path_repeatable = "<REPEATABLE_SQLS_PATH>"`
	`sql_path_finalize = "<FINALIZE_SQLS_PATH>"`
	`create_random_password = false`
	`master_username = "<REDSHIFT_MASTER_USERNAME>"`	

Operazione	Description	Competenze richieste

Distribuisci le risorse utilizzando Terraform.	1. Per prepararti al processo di distribuzione, usa il seguente comando per inizializzare Terraform all'interno del repository clonato.
	`terraform init`
	2. Per visualizzare in anteprima le modifiche che Terraform applicher à all'infrastruttura, utilizza il seguente comando per creare un piano di esecuzione.
	`terraform plan -var-file terraform.tfvars`
	3. Per effettuare il provisioning del cluster Amazon Redshift e delle risorse associate , utilizza il seguente comando per applicare il piano di esecuzione Terraform.
	`terraform apply -var-file terraform.tfvars`
		DevOps ingegnere

Operazione	Description	Competenze richieste

(Facoltativo) Esegui query SQL aggiuntive.	Il repository di esempio fornisce diverse query SQL a scopo dimostrativo. Per eseguire le tue query SQL, aggiungile alle seguenti cartelle:
	`/bootstrap`
	`/nonrepeatable`
	`/repeatable`
	`/finalize`	

Monitora l'esecuzione delle istruzioni SQL

Operazione	Description	Competenze richieste
Monitora la distribuzione delle istruzioni SQL.	Puoi monitorare i risultati delle esecuzioni SQL su un cluster Amazon Redshift. Per esempi di output che mostrano un'esecuzione SQL non riuscita e riuscita, consulta Esempi di istruzioni SQL in Informazioni aggiuntive.	DBA, ingegnere DevOps
Eliminare le risorse.	Per eliminare tutte le risorse distribuite da Terraform, esegui il seguente comando. `terraform destroy`	DevOps ingegnere

Convalida i risultati

Operazione	Description	Competenze richieste
Convalida i dati nel cluster Amazon Redshift.	1. Accedi a e apri la console Amazon Redshift. Console di gestione AWS 2. Dal menu di navigazione, scegliere Clusters (Cluster) . Scegli il nome del cluster pertinente nell'elenco. 3. Segui le istruzioni in Interrogazione di un database utilizzando l'editor di query Amazon Redshift v2 nella documentazione di Amazon Redshift.	DBA, AWS DevOps

Risorse correlate

AWS documentazione

- Cluster forniti da Amazon Redshift
- Risoluzione dei problemi per Amazon Redshift Data API

Altre risorse

- Comando: apply (documentazione Terraform)

Informazioni aggiuntive

Procedura dettagliata della soluzione

Per utilizzare la soluzione, è necessario organizzare le query SQL di Amazon Redshift in un modo specifico. Tutte le query SQL devono essere archiviate in file con estensione. `.sql`

Nell'esempio di codice fornito con questo modello, le query SQL sono organizzate nella seguente struttura di cartelle. È possibile modificare il codice (`sql-queries.tfesql-queries.py`) per utilizzarlo con qualsiasi struttura adatta al proprio caso d'uso specifico.

```
/bootstrap
     |- Any # of files
     |- Any # of sub-folders
/nonrepeatable
     |- Any # of files
     |- Any # of sub-folders
/repeatable
     /udf
          |- Any # of files
          |- Any # of sub-folders
     /table
          |- Any # of files
          |- Any # of sub-folders
     /view
          |- Any # of files
          |- Any # of sub-folders
     /stored-procedure
          |- Any # of files
          |- Any # of sub-folders
/finalize
     |- Any # of files
     |- Any # of sub-folders
```

Data la struttura delle cartelle precedente, durante la distribuzione del cluster Amazon Redshift, Terraform esegue le query nel seguente ordine:

1. `/bootstrap`
2. `/nonrepeatable`
3. `/repeatable`
4. `/finalize`

La `/repeatable` cartella contiene quattro sottocartelle:,, e. `/udf` `/table` `/view` `/stored-procedure` Queste sottocartelle indicano l'ordine in cui Terraform esegue le query SQL.

Lo script Python che esegue le query SQL è. `sql-queries.py` Innanzitutto, lo script legge tutti i file e le sottocartelle di una specifica directory di origine, ad esempio il parametro.

`sql_path_bootstrap` Quindi lo script esegue le query richiamando l'operazione dell'API Amazon ExecuteStatementRedshift. Potresti avere una o più query SQL in un file. Il seguente frammento di codice mostra la funzione Python che esegue istruzioni SQL archiviate in un file su un cluster Amazon Redshift.

```
def execute_sql_statement(filename, cluster_id, db_name, secret_arn, aws_region):
    """Execute SQL statements in a file"""
    redshift_client = boto3.client(
        'redshift-data', region_name=aws_region)
    contents = get_contents_from_file(filename),
    response = redshift_client.execute_statement(
        Sql=contents[0],
        ClusterIdentifier=cluster_id,
        Database=db_name,
        WithEvent=True,
        StatementName=filename,
        SecretArn=secret_arn
    )
    ...
```

Lo script Terraform `sql-queries.tf` crea le risorse terraform_data che richiamano lo script. `sql-queries.py` Esiste una `terraform_data` risorsa per ciascuna delle quattro cartelle:`/bootstrap`,, e. `/nonrepeatable` `/repeatable` `/finalize` Il seguente frammento di codice mostra la `terraform_data` risorsa che esegue le query SQL nella cartella. `/bootstrap`

```
locals {
  program               = "${path.module}/sql-queries.py"
  redshift_cluster_name = try(aws_redshift_cluster.this[0].id, null)
}

resource "terraform_data" "run_bootstrap_queries" {
  count      = var.create && var.run_nonrepeatable_queries && (var.sql_path_bootstrap !
= "") && (var.snapshot_identifier == null) ? 1 : 0
  depends_on = [aws_redshift_cluster.this[0]]

  provisioner "local-exec" {
    command = "python3 ${local.program} ${var.sql_path_bootstrap}
 ${local.redshift_cluster_name} ${var.database_name} ${var.redshift_secret_arn}
 ${local.aws_region}"
  }
}
```

È possibile controllare se eseguire queste query utilizzando le seguenti variabili. Se non desideri eseguire query in`sql_path_bootstrap`,, o `sql_path_nonrepeatable` `sql_path_repeatable`sql_path_finalize, imposta i relativi valori su. ""

```
  run_nonrepeatable_queries = true
  run_repeatable_queries    = true
  sql_path_bootstrap        = "src/redshift/bootstrap"
  sql_path_nonrepeatable    = "src/redshift/nonrepeatable"
  sql_path_repeatable       = "src/redshift/repeatable"
  sql_path_finalize         = "src/redshift/finalize"
```

Durante l'esecuzione`terraform apply`, Terraform considera la `terraform_data` risorsa aggiunta dopo il completamento dello script, indipendentemente dai risultati dello script. Se alcune query SQL hanno avuto esito negativo e desideri eseguirle nuovamente, puoi rimuovere manualmente la risorsa dallo stato Terraform ed eseguirle di nuovo. `terraform apply` Ad esempio, il comando seguente rimuove la `run_bootstrap_queries` risorsa dallo stato Terraform.

```
terraform state rm module.redshift.terraform_data.run_bootstrap_queries[0]
```

Il seguente esempio di codice mostra come la `run_repeatable_queries` risorsa monitora le modifiche nella `repeatable` cartella utilizzando l'hash sha256. Se un file all'interno della cartella viene aggiornato, Terraform contrassegna l'intera directory per un aggiornamento. Quindi, Terraform esegue nuovamente le query nella directory successiva. `terraform apply`

```
resource "terraform_data" "run_repeatable_queries" {
  count      = var.create_redshift && var.run_repeatable_queries &&
 (var.sql_path_repeatable != "") ? 1 : 0
  depends_on = [terraform_data.run_nonrepeatable_queries]

  # Continuously monitor and apply changes in the repeatable folder
  triggers_replace = {
    dir_sha256 = sha256(join("", [for f in fileset("${var.sql_path_repeatable}",
 "**") : filesha256("${var.sql_path_repeatable}/${f}")]))
  }

  provisioner "local-exec" {
    command = "python3 ${local.sql_queries} ${var.sql_path_repeatable}
 ${local.redshift_cluster_name} ${var.database_name} ${var.redshift_secret_arn}"
  }
}
```

Per perfezionare il codice, puoi implementare un meccanismo per rilevare e applicare le modifiche solo ai file che sono stati aggiornati all'interno della `repeatable` cartella, anziché applicare le modifiche a tutti i file indiscriminatamente.

Esempi di istruzioni SQL

L'output seguente mostra un'esecuzione SQL non riuscita, insieme a un messaggio di errore.

```
module.redshift.terraform_data.run_nonrepeatable_queries[0] (local-exec): Executing:
 ["/bin/sh" "-c" "python3 modules/redshift/sql-queries.py src/redshift/nonrepeatable
 testcluster-1 db1 arn:aws:secretsmanager:us-east-1:XXXXXXXXXXXX:secret:/redshift/
master_user/password-8RapGH us-east-1"]
module.redshift.terraform_data.run_nonrepeatable_queries[0] (local-exec):
 ----------------------------------------------------------------
module.redshift.terraform_data.run_nonrepeatable_queries[0] (local-exec): src/redshift/
nonrepeatable/table/admin/admin.application_family.sql
module.redshift.terraform_data.run_nonrepeatable_queries[0] (local-exec):
 ----------------------------------------------------------------
module.redshift.terraform_data.run_nonrepeatable_queries[0] (local-exec): Status:
 FAILED
module.redshift.terraform_data.run_nonrepeatable_queries[0] (local-exec): SQL execution
 failed.
module.redshift.terraform_data.run_nonrepeatable_queries[0] (local-exec): Error
 message: ERROR: syntax error at or near ")"
module.redshift.terraform_data.run_nonrepeatable_queries[0] (local-exec):   Position:
 244
module.redshift.terraform_data.run_nonrepeatable_queries[0]: Creation complete after 3s
 [id=ee50ba6c-11ae-5b64-7e2f-86fd8caa8b76]
```

L'output seguente mostra un'esecuzione SQL riuscita.

```
module.redshift.terraform_data.run_bootstrap_queries[0]: Provisioning with 'local-
exec'...
module.redshift.terraform_data.run_bootstrap_queries[0] (local-exec): Executing:
 ["/bin/sh" "-c" "python3 modules/redshift/sql-queries.py src/redshift/bootstrap
 testcluster-1 db1 arn:aws:secretsmanager:us-east-1:XXXXXXXXXXXX:secret:/redshift/
master_user/password-8RapGH us-east-1"]
module.redshift.terraform_data.run_bootstrap_queries[0] (local-exec):
 ----------------------------------------------------------------
module.redshift.terraform_data.run_bootstrap_queries[0] (local-exec): src/redshift/
bootstrap/db.sql
module.redshift.terraform_data.run_bootstrap_queries[0] (local-exec):
 ----------------------------------------------------------------
```

```
module.redshift.terraform_data.run_bootstrap_queries[0] (local-exec): Status: FINISHED
module.redshift.terraform_data.run_bootstrap_queries[0] (local-exec): SQL execution
 successful.
module.redshift.terraform_data.run_bootstrap_queries[0]: Creation complete after 2s
 [id=d565ef6d-be86-8afd-8e90-111e5ea4a1be]
```

Esporta i report di AWS Backup da tutta l'organizzazione in AWS Organizations come file CSV

Aromal Raj Jayarajan e Purushotham GK, Amazon Web Services

Riepilogo

Questo modello mostra come esportare i report sui job di AWS Backup da un'intera organizzazione in AWS Organizations come file CSV. La soluzione utilizza AWS Lambda e Amazon EventBridge per classificare i report sui job di AWS Backup in base al loro stato, il che può aiutare nella configurazione di automazioni basate sullo stato.

AWS Backup aiuta le organizzazioni a gestire e automatizzare centralmente la protezione dei dati tra i servizi AWS, nel cloud e in locale. Tuttavia, per i job di AWS Backup configurati all'interno di AWS Organizations, il reporting consolidato è disponibile solo nella Console di gestione AWS dell'account di gestione di ogni organizzazione. L'inclusione di questi report all'esterno dell'account di gestione può ridurre lo sforzo richiesto per il controllo e aumentare l'ambito delle automazioni, delle notifiche e degli avvisi.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un'organizzazione attiva in AWS Organizations che include almeno un account di gestione e un account membro
- AWS Backup configurato a livello di organizzazione in AWS Organizations (per ulteriori informazioni, consulta Automatizzare il backup centralizzato su larga scala tra i servizi AWS utilizzando AWS Backup sul blog AWS)
- Git, installato e configurato sul tuo computer locale

Limitazioni

La soluzione fornita in questo modello identifica le risorse AWS configurate solo per i job di AWS Backup. Il report non è in grado di identificare le risorse AWS che non sono configurate per il backup tramite AWS Backup.

Architecture

Stack tecnologico Target

- AWS Backup
- AWS CloudFormation
- Amazon EventBridge
- AWS Lambda
- AWS Security Token Service (AWS STS)
- Amazon Simple Storage Service (Amazon S3)
- AWS Identity and Access Management (IAM)

Architettura Target

Il diagramma seguente mostra un esempio di flusso di lavoro per esportare i report di lavoro di AWS Backup da un'organizzazione in AWS Organizations come file CSV.

Il diagramma mostra il flusso di lavoro seguente:

1. Una regola relativa agli EventBridge eventi pianificati richiama una funzione Lambda nell'account AWS membro (reporting).
2. La funzione Lambda utilizza quindi AWS STS per assumere un ruolo IAM con le autorizzazioni necessarie per connettersi all'account di gestione.
3. La funzione Lambda esegue quindi le seguenti operazioni:
   - Richiede il report consolidato sui lavori di AWS Backup dal servizio AWS Backup
   - Categorizza i risultati in base allo stato del job di AWS Backup
   - Converte la risposta in un file CSV
   - Carica i risultati in un bucket Amazon S3 nell'account di reporting all'interno di cartelle etichettate in base alla data di creazione

Tools (Strumenti)

Strumenti

- AWS Backup è un servizio completamente gestito che ti aiuta a centralizzare e automatizzare la protezione dei dati tra i servizi AWS, nel cloud e in locale.
- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account e regioni AWS.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, funzioni AWS Lambda, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altri account AWS.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Codice

Il codice per questo pattern è disponibile nel GitHub aws-backup-report-generatorrepository.

Best practice

- Best practice di sicurezza per Amazon S3 (Amazon S3 User Guide)
- Le migliori pratiche per lavorare con le funzioni di AWS Lambda (AWS Lambda Developer Guide)
- Le migliori pratiche per l'account di gestione (AWS Organizations User Guide)

Epiche

Implementa i componenti della soluzione

Operazione	Description	Competenze richieste
Clona il GitHub repository.	Clona il GitHub aws-backup-report-generatorrepository e eseguendo il seguente	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	comando in una finestra di terminale: `git clone https://github.com/aws-samples/aws-backup-report-generator.git` Per ulteriori informazioni, consulta Clonazione di un repository nei documenti. GitHub	

Operazione	Description	Competenze richieste
Implementa i componenti della soluzione nell'account AWS membro (reporting).	1. Nell'account membro (reporting), accedi alla Console di gestione AWS e quindi apri la CloudFormation console. 2. Scegliere Create stack (Crea stack), quindi scegliere Con nuove risorse (standard). 3. Nella pagina Crea stack, nella sezione Specificare il modello, scegli Carica un file modello. 4. Selezionare Choose file (Scegli file). Quindi, vai alla cartella principale del GitHub repository clonato sulla tua workstation locale e scegli template-reporting.yaml. 5. Scegliete Apri, quindi scegliete Avanti. 6. Nella pagina Specificare i dettagli dello stack, in Nome dello stack, inserisci un nome per lo stack. CloudFormation 7. Per ManagementAccountID, inserisci l'ID dell'account AWS per l'account di gestione della tua organizzazione in AWS Organizations.	DevOps ingegnere, AWS DevOps

Operazione	Description	Competenze richieste
	8. Scegli Next (Successivo). 9. Nella pagina Configure Stack Options, scegli Avanti. 10. Nella pagina Revisione , seleziona la casella di controllo per confermare di aver esaminato la configura zione. 11. Seleziona Crea stack. Lo stack mostra lo stato CREATE_COMPLETE quando i componenti della soluzione vengono distribui ti nell'account membro (reporting).	

Test della soluzione

Operazione	Description	Competenze richieste
Assicurati che la EventBridge regola venga eseguita prima del test.	Assicurati che la EventBrid ge regola venga eseguita aspettando almeno 24 ore o aumentando la frequenza dei report nel file CloudFormation template-reporting.yml del modello. Per aumentare la frequenza dei report 1. Apri il file template-reporting .yml nel repository clonato.	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	2. Nella regola degli eventi con l'ID logico '', trova il ''. LambdaSchedule ScheduleExpression 3. Modifica la chiave 'ScheduleExpression' in modo che includa un'espressione cron valida. Ad esempio, la seguente espressione cron pianifica l'esecuzione della regola dell'evento ogni cinque minuti: `''cron (* /5 * * * *)''`	

Operazione	Description	Competenze richieste
Controlla il bucket Amazon S3 per il report generato.	1. Nell'account membro (reporting), accedi alla Console di gestione AWS e quindi apri la CloudFormation console. 2. Nel riquadro Stacks, seleziona il nome dello stack che hai creato. Quindi, scegli la scheda Risorse. 3. Nel riquadro Risorse, nella colonna Logical ID, trova BackupReportS3Bucket. Quindi, apri il bucket Amazon S3 associato in una nuova scheda selezionando il link nella colonna ID fisico accanto all'ID logico. 4. <dd><Mon><yyyy>Assicurati che il bucket contenga un report generato nel seguente formato: BackupReports/// BackupReport- - - .csv <yyyy><mm><dd><BACKUP JOB STATUS>	AWS DevOps, DevOps ingegnere

Pulizia delle risorse

Operazione	Description	Competenze richieste
Eliminare i componenti della soluzione dall'account membro (di segnalazione).	1. Nell'account membro (reporting), apri il bucket Amazon S3 della soluzione . Per istruzioni, consulta i passaggi 2-4 del bucket Check the S3 per il report generato della sezione Test the solution di questo modello. 2. Elimina il contenuto del bucket e svuota il bucket. Per istruzioni, consulta Svuotare un bucket nella Guida per l'utente di Amazon S3. 3. Nell'account membro (reporting), accedi alla Console di gestione AWS e quindi apri la CloudFormation console. 4. Nel riquadro Stacks, seleziona la casella di controllo accanto al nome dello stack che hai creato. Quindi, scegli Elimina.	AWS DevOps, DevOps ingegnere
Elimina i componenti della soluzione dall'account di gestione.	1. Nell'account di gestione, accedi alla Console di gestione AWS e apri la CloudFormation console. 2. Nel riquadro Stacks, seleziona la casella di	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	controllo accanto al nome dello stack che hai creato. Quindi, scegli Elimina.	

Risorse correlate

- Tutorial: utilizzo di AWS Lambda con eventi pianificati (documentazione AWS Lambda)
- Creazione di eventi pianificati per eseguire funzioni AWS Lambda (SDK AWS per la documentazione) JavaScript
- Tutorial IAM: delega l'accesso tra account AWS utilizzando i ruoli IAM (documentazione IAM)
- Terminologia e concetti di AWS Organizations (documentazione di AWS Organizations)
- Creazione di piani di report utilizzando la console AWS Backup (documentazione AWS Backup)
- Crea un rapporto di audit (documentazione di AWS Backup)
- Creazione di report su richiesta (documentazione di AWS Backup)
- Cos'è AWS Backup? (documentazione di AWS Backup)
- Automatizza il backup centralizzato su larga scala tra i servizi AWS utilizzando AWS Backup (post sul blog AWS)

Esporta i tag per un elenco di EC2 istanze Amazon in un file CSV

Sida Ju e Pac Joonhyun, Amazon Web Services

Riepilogo

Questo modello mostra come esportare in modo programmatico i tag per un elenco di istanze Amazon Elastic Compute Cloud (Amazon EC2) in un file CSV.

Utilizzando lo script Python di esempio fornito, puoi ridurre il tempo necessario per esaminare e classificare le tue EC2 istanze Amazon in base a tag specifici. Ad esempio, puoi utilizzare lo script per identificare e classificare rapidamente un elenco di istanze che il tuo team di sicurezza ha segnalato per gli aggiornamenti software.

Prerequisiti e limitazioni

Prerequisiti

- Python 3 installato e configurato
- AWS Command Line Interface (AWS CLI) installata e configurata

Limitazioni

Lo script Python di esempio fornito in questo modello può cercare EC2 istanze Amazon solo in base ai seguenti attributi:

- Istanza IDs
- IPv4 Indirizzi privati
- IPv4 Indirizzi pubblici

Tools (Strumenti)

- Python è un linguaggio di programmazione per computer generico.
- virtualenv ti aiuta a creare ambienti Python isolati.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.

Repository di codice

Lo script Python di esempio per questo pattern è disponibile nel repository GitHub search-ec2 -. instances-export-tags

Epiche

Installa e configura i prerequisiti

Operazione	Description	Competenze richieste
Clona il GitHub repository.	**Note** Se ricevi errori durante l'esecuzione dei comandi dell'interfaccia a riga di comando di AWS, assicurati di utilizzare la versione più recente dell'interfaccia a riga di comando di AWS. Clona il instances-export-tags repository GitHub search-ec2- eseguendo il seguente comando Git in una finestra di terminale: `git clone https://github.com/aws-samples/search-ec2-instances-export-tags.git`	DevOps ingegnere
Installa e attiva virtualenv.	1. Installa virtualenv eseguendo il seguente comando:	DevOps ingegnere

Operazione	Description	Competenze richieste

	`python3 -m pip install virtualenv`
	2. Crea un nuovo ambiente virtuale eseguendo il seguente comando:
	`python3 -m venv env`
	3. Attiva il nuovo ambiente virtuale eseguendo il comando seguente:
	`source env/bin/activate`
	Per ulteriori informazioni, consulta la Guida per l'utente di virtualenv.	

Installare le dipendenze.	1. Apri la directory del codice eseguendo il seguente comando nel terminale:
	`cd search-ec2-instances-export-tags`
	2. Installa il `requirements.txt` file eseguendo il seguente comando pip:
	`pip3 install -r requirements.txt`
		DevOps ingegnere

Operazione	Description	Competenze richieste
Configura un profilo denominato AWS.	Se non l'hai già fatto, configura un profilo denominato AWS che includa le credenziali richieste per eseguire lo script. Per creare un profilo denominato, esegui il comando aws configure. Per ulteriori informazioni, consulta Using named profiles nella documentazione AWS CLI.	DevOps ingegnere

Configura ed esegui lo script Python

Operazione	Description	Competenze richieste
Crea il file di input.	Crea un file di input che contenga un elenco delle EC2 istanze Amazon per le quali desideri che lo script cerchi ed esporti i tag. Puoi elencare istanze IDs, IPv4 indirizzi privati o IPv4 indirizzi pubblici. **Important** Assicurati che ogni EC2 istanza Amazon sia elencata su una riga distinta nel file di input. Esempio di file di input	DevOps ingegnere

Operazione	Description	Competenze richieste
	i-0547c351bdfe85b9f 54.157.194.156 172.31.85.33 54.165.198.144 i-0b6223b5914111a4b 172.31.85.44 54.165.198.145 172.31.80.219 172.31.94.199	

Operazione	Description	Competenze richieste
Eseguire lo script Python.	Esegui lo script eseguendo il seguente comando nel terminale: `python search_instances.py -i INPUTFILE -o OUTPUTFILE -r REGION [-p PROFILE]` **Note** Sostituisci `INPUTFILE` con il nome del file di input. Sostituisci `OUTPUTFILE` con il nome che vuoi assegnare al file di output CSV. Sostituisci `REGION` con la regione AWS in cui si trovano EC2 le tue risorse Amazon. Se utilizzi un profilo denominato AWS, `PROFILE` sostituiscilo con il profilo denominato che stai utilizzando. Per ottenere un elenco dei parametri supportati e la loro descrizione, esegui il seguente comando:	DevOps ingegnere

Operazione	Description	Competenze richieste
	`python search_instances.py -h` Per ulteriori informazioni e per vedere un esempio di file di output, consultate il `README.md` file nel repository GitHub search-ec2-instances-export-tags	

Risorse correlate

- Configurazione dell'interfaccia a riga di comando di AWS (AWS CLI User Guide)

Genera un CloudFormation modello AWS contenente le regole gestite di AWS Config utilizzando Troposphere

Lucas Nation e Freddie Wilson, Amazon Web Services

Riepilogo

Molte organizzazioni utilizzano le regole gestite di AWS Config per valutare la conformità delle proprie risorse Amazon Web Services (AWS) rispetto alle best practice comuni. Tuttavia, la manutenzione di queste regole può richiedere molto tempo e questo modello ti aiuta a sfruttare Troposphere, una libreria Python, per generare e gestire regole gestite da AWS Config.

Il modello ti aiuta a gestire le regole gestite di AWS Config utilizzando uno script Python per convertire un foglio di calcolo di Microsoft Excel contenente le regole gestite da AWS in un modello AWS. CloudFormation Troposphere funge da infrastruttura come codice (IaC) e ciò significa che puoi aggiornare il foglio di calcolo Excel con regole gestite, invece di utilizzare un file in formato JSON o YAML. Utilizza quindi il modello per avviare uno CloudFormation stack AWS che crea e aggiorna le regole gestite nel tuo account AWS.

Il CloudFormation modello AWS definisce ogni regola gestita di AWS Config utilizzando il foglio di calcolo di Excel e ti aiuta a evitare di creare manualmente singole regole nella Console di gestione AWS. Lo script imposta per impostazione predefinita i parametri di ogni regola gestita su un dizionario vuoto e i valori predefiniti dell'ambito da. *`ComplianceResourceTypes`* *`THE_RULE_IDENTIFIER.template file`* Per ulteriori informazioni sull'identificatore della regola, consulta Creazione di regole gestite AWS Config con modelli AWS nella CloudFormation documentazione di AWS Config.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Familiarità con l'uso di CloudFormation modelli AWS per creare regole gestite AWS Config. Per ulteriori informazioni su questo argomento, consulta Creazione di regole gestite AWS Config con CloudFormation modelli AWS nella documentazione di AWS Config.
- Python 3, installato e configurato. Per ulteriori informazioni su questo argomento, consulta la documentazione di Python.
- Un ambiente di sviluppo integrato (IDE) esistente.

- Identifica le tue unità organizzative (OUs) in una colonna del foglio di calcolo `excel_config_rules.xlsx` Excel di esempio (allegato).

Epiche

Personalizza e configura le regole gestite di AWS Config

Operazione	Description	Competenze richieste
Aggiorna il foglio di calcolo Excel di esempio.	Scarica il foglio di calcolo `excel_config_rules .xlsx` Excel di esempio (allegato) ed etichetta come `Implemented` le regole gestite di AWS Config che desideri utilizzare. Le regole contrassegnate come `Implemented` verranno aggiunte al CloudFormation modello AWS.	Developer

(Facoltativo) Aggiorna il file config_rules_params.json con i parametri delle regole AWS Config.	Alcune regole gestite di AWS Config richiedono parametri e devono essere passate allo script Python come file JSON utilizzando l'opzione. `--param-file` Ad esempio, la regola `access-keys-rotated` gestita utilizza il seguente parametro: `maxAccessKeyAge`
	
	`{`
	`"access-keys-rotated": {`
		Developer

Operazione	Description	Competenze richieste

	<pre>"InputPar
	ameters": {
	
	"maxAccessKeyAge": 90
	}
	}
	}</pre>
	In questo parametro di esempio, `maxAccess KeyAge` è impostato su 90 giorni. Lo script legge il file dei parametri e aggiunge `InputParameters` quello che trova.	

Operazione	Description	Competenze richieste
(Facoltativo) Aggiorna il file config_rules_params.json con AWS Config. ComplianceResourceTypes	Per impostazione predefinita, lo script Python recupera i modelli definiti da `ComplianceResourceTypes` AWS. Se desideri sovrascrivere l'ambito di una specifica regola gestita di AWS Config, devi passarla allo script Python come file JSON utilizzando l'opzione. `--param-file` Ad esempio, il seguente codice di esempio mostra come il `ComplianceResourceTypes` for `ec2-volume-inuse-check` è impostato sulla lista: `["AWS::EC2::Volume"]` `{ "ec2-volume-inuse-check": { "Scope": { "ComplianceResourceTypes": [ "AWS::EC2::Volume" ] } } }`	Developer

Esegui lo script Python

Operazione	Description	Competenze richieste
Installa i pacchetti pip dal file requirements.txt.	Scarica il `requirements.txt` file (allegato) ed esegui il seguente comando nel tuo IDE per installare i pacchetti Python: `pip3 install -r requirements.txt`	Developer
Eseguire lo script Python.	1. Scarica il `aws_config_rules.py` file (allegato) sul tuo computer locale. 2. **Note** Esegui il `python3 aws_config_rules.py --ou <OU_NAME>` comando.: `--ou` definisce quale colonna OU scegliere nel foglio di calcolo Excel. È inoltre possibile aggiungere i seguenti parametri opzionali: • `--config-rule-option` — Definisce le regole da scegliere dal foglio di	Developer

Operazione	Description	Competenze richieste
	calcolo Excel. L'impostazione predefinita è il parametro. `Implemented` • `--excel-file` — Il percorso del foglio di calcolo Excel. Il valore predefinito è `aws_config_rules.xlsx` . • `--param-file` — Il percorso del file JSON dei parametri. Il valore predefinito è `config_rules_params.json` . • `--max-execution-frequency` — Definisce la frequenza con cui vengono valutate le regole gestite di AWS Config. Le scelte sono`One_Hour`,, `Three_Hours` `Six_Hours` `Twelve_Hours` , o. `TwentyFour_Hours` Il valore predefinito è `TwentyFour_Hours` .	

Implementa le regole gestite di AWS Config

Operazione	Description	Competenze richieste
Avvia lo CloudFormation stack AWS.	1. Accedi alla Console di gestione AWS, apri la CloudFormation console AWS e scegli Create stack.	Developer

Operazione	Description	Competenze richieste
	2. Nella pagina Specificare il modello, scegli Carica un file modello, quindi carica il tuo CloudFormation modello AWS. 3. Specificate un nome per lo stack e poi scegliete Avanti. 4. Specificate i tag, quindi scegliete Avanti. 5. Seleziona Crea stack.	

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Offri alle istanze di SageMaker notebook l'accesso temporaneo a un CodeCommit repository in un altro account AWS

Helge Aufderheide, Amazon Web Services

Riepilogo

Questo modello mostra come concedere alle istanze e agli utenti di SageMaker notebook Amazon l'accesso temporaneo a un CodeCommit repository AWS che si trova in un altro account AWS. Questo modello mostra anche come è possibile concedere autorizzazioni granulari per azioni specifiche che ciascuna entità può eseguire su ciascun repository.

Organizations spesso archivia i CodeCommit repository in un account AWS diverso da quello che ospita il loro ambiente di sviluppo. Questa configurazione multi-account aiuta a controllare l'accesso ai repository e riduce il rischio che vengano eliminati accidentalmente. Per concedere queste autorizzazioni su più account, è consigliabile utilizzare i ruoli AWS Identity and Access Management (IAM). Quindi, le identità IAM predefinite in ogni account AWS possono assumere temporaneamente i ruoli per creare una catena di fiducia controllata tra gli account.

> **Note**
>
> Puoi applicare una procedura simile per concedere ad altre identità IAM l'accesso a un repository da più account. CodeCommit Per ulteriori informazioni, consulta Configurare l'accesso tra account a un CodeCommit repository AWS utilizzando i ruoli nella AWS CodeCommit User Guide.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo con un CodeCommit repository (account A)
- Un secondo account AWS attivo con un'istanza SageMaker notebook (account B)
- Un utente AWS con autorizzazioni sufficienti per creare e modificare ruoli IAM nell'account A
- Un secondo utente AWS con autorizzazioni sufficienti per creare e modificare i ruoli IAM nell'account B

Architecture

Il diagramma seguente mostra un esempio di flusso di lavoro per concedere a un'istanza di SageMaker notebook e agli utenti di un account AWS l'accesso multiaccount a un repository: CodeCommit

Il diagramma mostra il flusso di lavoro seguente:

1. Il ruolo utente AWS e il ruolo dell'istanza SageMaker notebook nell'account B presuppongono un profilo denominato.
2. La politica di autorizzazione del profilo denominato specifica un ruolo di CodeCommit accesso nell'account A che il profilo assume successivamente.
3. La politica di fiducia del ruolo di CodeCommit accesso nell'account A consente al profilo denominato nell'account B di assumere il CodeCommit ruolo di accesso.
4. La politica di autorizzazione IAM del CodeCommit repository nell'account A consente al ruolo di CodeCommit accesso di accedere al CodeCommit repository.

Stack tecnologico

- CodeCommit
- Git
- IAM
- pip
- SageMaker

Tools (Strumenti)

- AWS CodeCommit è un servizio di controllo delle versioni che consente di archiviare e gestire archivi Git in modo privato, senza dover gestire il proprio sistema di controllo del codice sorgente.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- Git è un sistema distribuito di controllo delle versioni per tenere traccia delle modifiche nel codice sorgente durante lo sviluppo del software.

- git-remote-codecommitè un'utilità che consente di inviare ed estrarre codice dai CodeCommit repository estendendo Git.
- pip è l'installatore di pacchetti per Python. Puoi usare pip per installare pacchetti dal Python Package Index e da altri indici.

Best practice

Quando imposti le autorizzazioni con le policy IAM, assicurati di concedere solo le autorizzazioni necessarie per eseguire un'attività. Per ulteriori informazioni, consulta Applicazione delle autorizzazioni del privilegio minimo nella documentazione di IAM.

Quando implementi questo modello, assicurati di fare quanto segue:

- Verifica che i principi IAM dispongano solo delle autorizzazioni necessarie per eseguire azioni specifiche e necessarie all'interno di ciascun repository. Ad esempio, si consiglia di consentire ai principi IAM approvati di inviare e unire le modifiche a rami specifici del repository, ma di richiedere le unioni solo ai rami protetti.
- Verifica che ai principi IAM siano assegnati ruoli IAM diversi in base ai rispettivi ruoli e responsabilità per ciascun progetto. Ad esempio, uno sviluppatore avrà autorizzazioni di accesso diverse da quelle di un release manager o di un amministratore AWS.

Epiche

Configura i ruoli IAM

Operazione	Description	Competenze richieste
Configura il ruolo di CodeCommit accesso e la politica delle autorizzazioni.	**Note** Per automatizzare il processo di configurazione manuale documentato in questa epopea, puoi utilizzare un modello AWS. CloudFormation	Informazioni generali su AWS, AWS DevOps

Operazione	Description	Competenze richieste
	Nell'account che contiene il CodeCommit repository (account A), procedi come segue: 1. Crea un ruolo IAM che possa essere assunto dal ruolo dell'istanza SageMaker notebook nell'account B. 2. Crea una policy IAM che conceda l'accesso al repository e allega la policy al ruolo. Solo a scopo di test, scegli la policy gestita da AWSCodeCommitPowerUserAWS. Questa policy concede tutte le CodeCommit autorizzazioni tranne la possibilità di eliminare le risorse. 3. Modifica la politica di fiducia del ruolo in modo che l'account B sia elencato come entità attendibile. **Important** Prima di spostare questa configurazione nell'ambiente di produzione, è consigliabile scrivere una policy IAM	

Operazione	Description	Competenze richieste
	personalizzata che applichi le autorizza zioni con privilegi minimi. Per ulteriori informazioni, consulta la sezione Informazi oni aggiuntive di questo modello.	

Operazione	Description	Competenze richieste
Concedi al ruolo dell'istanza SageMaker notebook nell'account B le autorizzazioni per assumere il ruolo di CodeCommit accesso nell'account A.	Nell'account che contiene il ruolo IAM dell'istanza SageMaker notebook (account B), procedi come segue: 1. Crea una policy IAM che consenta a un ruolo o utente IAM di assumere il ruolo di CodeCommit accesso nell'account A. Esempio di policy di autorizzazione IAM che consente a un ruolo o utente IAM di assumere un ruolo su più account `{ "Version": "2012-10-17", "Statement": [ { "Sid": "VisualEditor0", "Effect": "Allow", "Action": "sts:AssumeRole", "Resource": "arn:aws:iam:::accountA_ID:role/accountArole_ID" } }` 2. Allega la policy al ruolo dell'istanza del tuo SageMaker notebook nell'account B.	Informazioni generali su AWS, AWS DevOps

Operazione	Description	Competenze richieste
	3. Fai in modo che il ruolo dell'istanza SageMaker notebook nell'account B assuma il ruolo di CodeCommit accesso nell'account A. **Note** Per visualizzare l'Amazon Resource Name (ARN) del tuo repository, consulta CodeCommit Visualizza i dettagli del repository nella AWS CodeCommit User Guide.	

Configura l'istanza del tuo SageMaker notebook nell'account B

Operazione	Description	Competenze richieste
Configura un profilo utente sull'istanza del SageMaker notebook AWS per assumere il ruolo nell'account A.	**Important** Assicurati di avere installata la versione più recente di AWS Command Line Interface (AWS CLI).	Informazioni generali su AWS, AWS DevOps

Operazione	Description	Competenze richieste

	Nell'account che contiene l'istanza del SageMaker notebook (account B), procedi come segue:
	
	1. Accedi alla console di gestione AWS e apri la console SageMaker .
	2. Accedi all'istanza del tuo SageMaker notebook. Si apre l'interfaccia di Jupyter.
	3. Scegli Nuovo, quindi scegli Terminale. Si apre una nuova finestra di terminale nell'ambiente Jupyter.
	4. Vai al file SageMaker ~/.aws/config dell'istanza del notebook. Quindi, aggiungi un profilo utente al file inserendo la seguente dichiarazione:
	
	`------.aws/config-`
	`-------------`
	`[profile remoterep`
	`ouser]`
	`role_arn = arn:aws:i`
	`am::<ID of Account`
	`A>:role/<rolename>`
	`role_session_name =`
	`remoteaccesssession`
	`region = eu-west-1`
	`credential_source =`
	`Ec2InstanceMetadata`	

Operazione	Description	Competenze richieste
	`----------` `----------`	
Installa l'utilità git-remote-codecommit.	Segui le istruzioni nella Fase 2: Installazione git-remote-codecommit nella AWS CodeCommit User Guide.	Data Scientist

Accedi al repository

Operazione	Description	Competenze richieste
Accedi al CodeCommit repository utilizzando i comandi Git o SageMaker.	Per usare Git I principali IAM che assumono il ruolo dell'istanza SageMaker notebook nell'account B possono ora eseguire comandi Git per accedere al CodeCommit repository nell'account A. Ad esempio, gli utenti possono eseguire comandi come `git clonegit pull`, e. `git push` Per istruzioni, consulta Connect to an AWS CodeCommit repository nella AWS CodeCommit User Guide. Per informazioni su come usare Git con CodeCommit, consulta Getting started with	Git, console bash

Operazione	Description	Competenze richieste
	AWS CodeCommit nella AWS CodeCommit User Guide. Da usare SageMaker Per usare Git dalla SageMaker console, devi consentire a Git di recuperare le credenziali dal tuo CodeCommit repository. Per istruzioni, consulta Associare un CodeCommit repository in un altro account AWS a un'istanza notebook nella SageMaker documentazione.	

Risorse correlate

- Configurare l'accesso tra account a un CodeCommit repository AWS utilizzando i ruoli (documentazione CodeCommit AWS)
- Tutorial IAM: delega l'accesso tra account AWS utilizzando i ruoli IAM (documentazione IAM)

Informazioni aggiuntive

Limitazione delle CodeCommit autorizzazioni a azioni specifiche

Per limitare le azioni che un responsabile IAM può intraprendere nel CodeCommit repository, modifica le azioni consentite nella CodeCommit policy di accesso.

Per ulteriori informazioni sulle operazioni CodeCommit API, consulta il riferimento CodeCommit alle autorizzazioni nella AWS CodeCommit User Guide.

> **Note**
>
> Puoi anche modificare la policy gestita di AWSCodeCommitPowerUserAWS per adattarla al tuo caso d'uso.

Limitazione delle CodeCommit autorizzazioni a repository specifici

Per creare un ambiente multitenant in cui più di un repository di codice siano accessibili solo a utenti specifici, procedi come segue:

1. Crea più ruoli di CodeCommit accesso nell'account A. Quindi, configura la politica di fiducia di ciascun ruolo di accesso per consentire a utenti specifici dell'account B di assumere il ruolo.
2. Limita gli archivi di codice che ogni ruolo può assumere aggiungendo una condizione «Risorsa» alla politica di ciascun ruolo di CodeCommit accesso.

Esempio di condizione «Resource» che limita l'accesso di un principale IAM a un repository specifico CodeCommit

```
"Resource" : [<REPOSITORY_ARN>,<REPOSITORY_ARN> ]
```

> **Note**
>
> Per aiutare a identificare e differenziare più repository di codice nello stesso account AWS, puoi assegnare prefissi diversi ai nomi dei repository. Ad esempio, puoi denominare repository di codice con prefissi che si allineano a diversi gruppi di sviluppatori, come myproject-subproject1-repo1 e myproject-subproject2-repo1. Quindi, puoi creare un ruolo IAM per ogni gruppo di sviluppatori in base ai prefissi assegnati. Ad esempio, puoi creare un ruolo denominato myproject-subproject1-repoaccess e concedergli l'accesso a tutti gli archivi di codice che includono il prefisso myproject-subproject1.

Esempio di condizione «Resource» che si riferisce a un ARN di un repository di codice che include un prefisso specifico

```
"Resource" : arn:aws:codecommit:<region>:<account-id>:myproject-subproject1-*
```

Implementa una strategia di ramificazione GitHub Flow per ambienti con più account DevOps

Mike Stephens e Abhilash Vinod, Amazon Web Services

Riepilogo

Quando si gestisce un repository di codice sorgente, diverse strategie di ramificazione influiscono sui processi di sviluppo e rilascio del software utilizzati dai team di sviluppo. Esempi di strategie di ramificazione comuni includono Trunk, GitHub Flow e Gitflow. Queste strategie utilizzano rami diversi e le attività svolte in ciascun ambiente sono diverse. Organizations che stanno implementando DevOps processi trarrebbero vantaggio da una guida visiva per aiutarle a comprendere le differenze tra queste strategie di ramificazione. L'utilizzo di questa immagine nell'organizzazione aiuta i team di sviluppo ad allineare il proprio lavoro e a seguire gli standard organizzativi. Questo modello fornisce questa immagine e descrive il processo di implementazione di una strategia di ramificazione GitHub Flow nell'organizzazione.

Questo modello fa parte di una serie di documentazione sulla scelta e l'implementazione di strategie di DevOps ramificazione per organizzazioni con più membri. Account AWS Questa serie è progettata per aiutarti ad applicare la strategia e le migliori pratiche corrette sin dall'inizio, per semplificare la tua esperienza nel cloud. GitHub Flow è solo una delle possibili strategie di ramificazione che l'organizzazione può utilizzare. Questa serie di documentazione copre anche i modelli di ramificazione Trunk e Gitflow. Se non l'hai già fatto, ti consigliamo di leggere Scelta di una strategia di ramificazione Git per DevOps ambienti multi-account prima di implementare le linee guida di questo modello. Utilizza la due diligence per scegliere la strategia di ramificazione giusta per la tua organizzazione.

Questa guida fornisce un diagramma che mostra come un'organizzazione potrebbe implementare la GitHub strategia Flow. Si consiglia di consultare la AWS DevOps Well-Architected Guidance per esaminare le best practice. Questo modello include attività, passaggi e restrizioni consigliati per ogni fase del DevOps processo.

Prerequisiti e limitazioni

Prerequisiti

- Git, installato. Viene utilizzato come strumento di archiviazione del codice sorgente.
- Draw.io, installato. Questa applicazione viene utilizzata per visualizzare e modificare il diagramma.

Architecture

Architettura Target

Il diagramma seguente può essere usato come un quadrato di Punnett (Wikipedia). Allineate i rami sull'asse verticale con gli AWS ambienti sull'asse orizzontale per determinare quali azioni eseguire in ogni scenario. I numeri indicano la sequenza delle azioni nel flusso di lavoro. In questo esempio si passa da una `feature` filiale all'implementazione in produzione.

Per ulteriori informazioni sugli ambienti e sui Account AWS rami di un approccio GitHub Flow, vedi Scelta di una strategia di ramificazione Git per ambienti con più account DevOps .

Automazione e scalabilità

L'integrazione continua e la distribuzione continua (CI/CD) is the process of automating the software release lifecycle. It automates much or all of the manual processes traditionally required to get new code from an initial commit into production. A CI/CD pipeline encompasses the sandbox, development, testing, staging, and production environments. In each environment, the CI/CD pipeline provisions any infrastructure that is needed to deploy or test the code. By using CI/CD, development teams can make changes to code that are then automatically tested and deployed. CI/CDpipeline) forniscono inoltre governance e barriere ai team di sviluppo, applicando coerenza, standard, best practice e livelli minimi di accettazione per l'accettazione e l'implementazione delle funzionalità. Per ulteriori informazioni, consulta Practicing Continuous Integration and Continuous Delivery su. AWS

AWS offre una suite di servizi per sviluppatori progettati per aiutarti a creare CI/CD pipeline. Ad esempio, AWS CodePipelineè un servizio di distribuzione continua completamente gestito che consente di automatizzare le pipeline di rilascio per aggiornamenti rapidi e affidabili di applicazioni e infrastrutture. AWS CodeBuildcompila il codice sorgente, esegue test e produce ready-to-deploy pacchetti software. Per ulteriori informazioni, consulta Developer Tools on AWS.

Tools (Strumenti)

AWS servizi e strumenti

AWS fornisce una suite di servizi per sviluppatori che è possibile utilizzare per implementare questo modello:

- AWS CodeArtifactè un servizio di repository di artefatti gestito e altamente scalabile che consente di archiviare e condividere pacchetti software per lo sviluppo di applicazioni.

- AWS CodeBuildè un servizio di compilazione completamente gestito che consente di compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS CodeDeployautomatizza le distribuzioni su Amazon Elastic Compute Cloud (Amazon EC2) o istanze AWS Lambda , funzioni o servizi Amazon Elastic Container Service (Amazon ECS) in locale.
- AWS CodePipelineti aiuta a modellare e configurare rapidamente le diverse fasi di un rilascio del software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.

Altri strumenti

- Draw.io Desktop è un'applicazione per creare diagrammi di flusso e diagrammi. Il repository di codice contiene modelli in formato.drawio per Draw.io.
- Figma è uno strumento di progettazione online progettato per la collaborazione. Il repository di codice contiene modelli in formato.fig per Figma.

Deposito di codice

Questo file sorgente per il diagramma in questo modello è disponibile nel GitHub repository Git Branching Strategy for GitHub Flow. Include file nei formati PNG, draw.io e Figma. È possibile modificare questi diagrammi per supportare i processi dell'organizzazione.

Best practice

Segui le best practice e i consigli contenuti in AWS DevOps Well-Architected Guidance e Choosing a Git branching strategy per ambienti multi-account. DevOps Questi consentono di implementare efficacemente lo sviluppo GitHub basato su Flow, promuovere la collaborazione, migliorare la qualità del codice e semplificare il processo di sviluppo.

Epiche

Revisione dei GitHub flussi di lavoro Flow

Operazione	Description	Competenze richieste
Esamina la procedura standard di GitHub Flow.	1. Nell'ambiente sandbox, lo sviluppatore crea	DevOps ingegnere

Operazione	Description	Competenze richieste
	un `feature` ramo dal `main` ramo e utilizza lo schema di denominazione. `feature/<ticket>_<initials>_<short description>` 2. Lo sviluppatore aggiunge uno o più commit al `feature` ramo, ognuno dei quali rappresenta una modifica o un miglioramento discreto. 3. Lo sviluppatore apre una richiesta di unione (MR) per unire le modifiche nel ramo. `main` Questo avvia un processo di revisione. 4. Durante il processo di revisione, gli sviluppatori discutono delle modifiche al codice e forniscono feedback. L'obiettivo è garantire che le modifiche siano di alta qualità e soddisfino gli standard del progetto. 5. Dopo che lo sviluppat ore ha creato la richiesta di unione, viene avviato un processo di compilazi one automatizzato che distribuisce le modifiche nel `feature` ramo nell'ambi ente di sviluppo.	

Operazione	Description	Competenze richieste
	6. I test automatici verificano l'integrità e la qualità delle modifiche incluse nella richiesta di unione. Per completare la richiesta di unione sono necessari una compilazione, una distribuzione e un test di successo. 7. Una volta completato il processo di revisione, le modifiche vengono unite alla `main` filiale. 8. Un approvatore approva manualmente la distribuzione degli elementi di rilascio nell'ambiente di test. 9. Un approvatore approva manualmente la distribuzione degli elementi di rilascio nell'ambiente di staging. 10. Un approvatore approva manualmente la distribuzione degli elementi di rilascio nell'ambiente di produzione.	

Operazione	Description	Competenze richieste
Esamina il processo bugfix GitHub Flow.	1. Lo sviluppatore crea un `bugfix` ramo dal `main` ramo e utilizza lo schema di denominazione. `bugfix/<ticket number>_<developer initials>_<descriptor>` 2. Lo sviluppatore risolve il problema, esegue la correzione e crea il ramo. `bugfix` 3. Lo sviluppatore apre una richiesta di unione per unire il ramo nel `bugfix` ramo. `main` Questo avvia un processo di revisione. 4. Durante il processo di revisione, gli sviluppatori discutono delle modifiche al codice e forniscono feedback. 5. Dopo il completamento e l'approvazione della revisione, lo sviluppatore completa la richiesta di fusione della `bugfix` filiale nella `main` filiale. 6. Un approvatore approva manualmente l'impleme ntazione degli elementi di rilascio in ambienti superiori .	DevOps ingegnere

Operazione	Description	Competenze richieste
Esamina il processo di hotfix GitHub Flow.	GitHub Flow è progettato per consentire la distribuzione continua, in cui le modifiche al codice vengono implement ate frequentemente e in modo affidabile in ambienti superiori. La chiave è che ogni `feature` filiale è implementabile in qualsiasi momento. `Hotfix`le filiali, che sono simili alle nostre `feature` `bugfix` filiali, possono seguire lo stesso processo di entrambe le altre filiali. Tuttavia, data la loro urgenza, gli hotfix hanno in genere una priorità più elevata. A seconda delle politiche del team e dell'imme diatezza della situazione, alcune fasi del processo potrebbero essere accelerat e. Ad esempio, le revisioni del codice per gli hotfix potrebber o essere rapide. Pertanto, sebbene il processo di hotfix sia parallelo al processo relativo a funzionalità o bugfix, l'urgenza relativa agli hotfix può giustificare modifiche nell'aderenza procedurale. È fondamentale stabilire linee guida sulla gestione degli hotfix per garantire	DevOps ingegnere

Operazione	Description	Competenze richieste
	che vengano gestiti in modo efficiente e sicuro.	

risoluzione dei problemi

Problema	Soluzione
conflitti tra filiali	Un problema comune che può verificarsi con il modello GitHub Flow è rappresentato dalla necessità di applicare un hotfix in produzion e `featurebugfix`, mentre una modifica corrispondente deve avvenire in un `hotfix` ramo in cui vengono modificate le stesse risorse. Si consiglia di unire frequentemente le modifiche provenienti dai `main` rami inferiori per evitare conflitti significativi durante l'unione a. `main`
Maturità del team	GitHub Flow incoraggia le implementazioni quotidiane in ambienti superiori, adottando una vera integrazione continua e una distribuzione continua (CI/CD). È fondamentale che il team abbia la maturità ingegneristica necessaria per creare funzionalità e creare test di automazione per esse. Il team deve eseguire una revisione esaustiva della richiesta di unione prima dell'approvazione delle modifiche. Ciò favorisce una solida cultura ingegneristica che promuove la qualità, la responsabilità e l'efficienza nel processo di sviluppo.

Risorse correlate

Questa guida non include la formazione per Git; tuttavia, ci sono molte risorse di alta qualità disponibili su Internet se hai bisogno di questa formazione. Ti consigliamo di iniziare dal sito di documentazione di Git.

Le seguenti risorse possono aiutarti nel tuo percorso di ramificazione di GitHub Flow in. Cloud AWS

AWS DevOps guida

- AWS DevOps Linee guida
- AWS Architettura di riferimento della pipeline di distribuzione
- Che cos'è DevOps?
- DevOps resources

GitHub Guida al flusso

- GitHub Tutorial di avvio rapido di Flow () GitHub
- Perché GitHub Flow?

Altre risorse

- Metodologia delle app a dodici fattori (12factor.net)

Implementa una strategia di ramificazione Gitflow per ambienti con più account DevOps

Mike Stephens, Stephen DiCato, Abhilash Vinod e Tim Wondergem, Amazon Web Services

Riepilogo

Quando si gestisce un repository di codice sorgente, diverse strategie di ramificazione influiscono sui processi di sviluppo e rilascio del software utilizzati dai team di sviluppo. Esempi di strategie di ramificazione comuni includono Trunk, Gitflow e Flow. GitHub Queste strategie utilizzano rami diversi e le attività svolte in ciascun ambiente sono diverse. Organizations che stanno implementando DevOps processi trarrebbero vantaggio da una guida visiva per aiutarle a comprendere le differenze tra queste strategie di ramificazione. L'utilizzo di questa immagine nell'organizzazione aiuta i team di sviluppo ad allineare il proprio lavoro e a seguire gli standard organizzativi. Questo modello fornisce questa immagine e descrive il processo di implementazione di una strategia di ramificazione Gitflow nella vostra organizzazione.

Questo modello fa parte di una serie di documentazione sulla scelta e l'implementazione di strategie di DevOps ramificazione per organizzazioni con più membri. Account AWS Questa serie è progettata per aiutarti ad applicare la strategia e le migliori pratiche corrette sin dall'inizio, per semplificare la tua esperienza nel cloud. Gitflow è solo una possibile strategia di ramificazione che la tua organizzazione può utilizzare. Questa serie di documentazione copre anche i modelli di ramificazione Trunk e GitHub Flow. Se non l'hai già fatto, ti consigliamo di leggere Scelta di una strategia di ramificazione Git per DevOps ambienti multi-account prima di implementare le linee guida di questo modello. Utilizza la due diligence per scegliere la strategia di ramificazione giusta per la tua organizzazione.

Questa guida fornisce un diagramma che mostra come un'organizzazione potrebbe implementare la strategia Gitflow. Si consiglia di consultare la AWS DevOps Well-Architected Guidance per esaminare le best practice. Questo modello include attività, passaggi e restrizioni consigliati per ogni fase del DevOps processo.

Prerequisiti e limitazioni

Prerequisiti

- Git, installato. Viene utilizzato come strumento di archiviazione del codice sorgente.
- Draw.io, installato. Questa applicazione viene utilizzata per visualizzare e modificare il diagramma.
- (Facoltativo) Plugin Gitflow, installato.

Architecture

Architettura Target

Il diagramma seguente può essere usato come un quadrato di Punnett (Wikipedia). Allineate i rami sull'asse verticale con gli AWS ambienti sull'asse orizzontale per determinare quali azioni eseguire in ogni scenario. I numeri indicano la sequenza delle azioni nel flusso di lavoro. In questo esempio si passa da un feature branch all'implementazione in produzione.

Per ulteriori informazioni sugli ambienti e sui Account AWS rami in un approccio Gitflow, consulta Scelta di una strategia di ramificazione Git per ambienti con più account. DevOps

Automazione e scalabilità

L'integrazione continua e la distribuzione continua (CI/CD) is the process of automating the software release lifecycle. It automates much or all of the manual processes traditionally required to get new code from an initial commit into production. A CI/CD pipeline encompasses the sandbox, development, testing, staging, and production environments. In each environment, the CI/CD pipeline provisions any infrastructure that is needed to deploy or test the code. By using CI/CD, development teams can make changes to code that are then automatically tested and deployed. CI/CDpipeline) forniscono inoltre governance e barriere ai team di sviluppo, applicando coerenza, standard, best practice e livelli minimi di accettazione per l'accettazione e l'implementazione delle funzionalità. Per ulteriori informazioni, consulta Practicing Continuous Integration and Continuous Delivery su. AWS

AWS offre una suite di servizi per sviluppatori progettati per aiutarti a creare CI/CD pipeline. Ad esempio, AWS CodePipelineè un servizio di distribuzione continua completamente gestito che consente di automatizzare le pipeline di rilascio per aggiornamenti rapidi e affidabili di applicazioni e infrastrutture. AWS CodeBuildcompila il codice sorgente, esegue test e produce ready-to-deploy pacchetti software. Per ulteriori informazioni, consulta Developer Tools on AWS.

Tools (Strumenti)

AWS servizi e strumenti

AWS fornisce una suite di servizi per sviluppatori che è possibile utilizzare per implementare questo modello:

- AWS CodeArtifactè un servizio di repository di artefatti gestito e altamente scalabile che consente di archiviare e condividere pacchetti software per lo sviluppo di applicazioni.

- AWS CodeBuildè un servizio di compilazione completamente gestito che consente di compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS CodeDeployautomatizza le distribuzioni su Amazon Elastic Compute Cloud (Amazon EC2) o su istanze AWS Lambda , funzioni o servizi Amazon Elastic Container Service (Amazon ECS) in locale.
- AWS CodePipelineti aiuta a modellare e configurare rapidamente le diverse fasi di un rilascio del software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.

Altri strumenti

- Draw.io Desktop è un'applicazione per creare diagrammi di flusso e diagrammi. Il repository di codice contiene modelli in formato.drawio per Draw.io.
- Figma è uno strumento di progettazione online progettato per la collaborazione. Il repository di codice contiene modelli in formato.fig per Figma.
- (Facoltativo) Il plugin Gitflow è una raccolta di estensioni Git che forniscono operazioni di repository di alto livello per il modello di ramificazione Gitflow.

Archivio di codice

Questo file sorgente per il diagramma in questo modello è disponibile nel GitHub repository Git Branching Strategy for GitFlow. Include file nei formati PNG, draw.io e Figma. È possibile modificare questi diagrammi per supportare i processi dell'organizzazione.

Best practice

Segui le migliori pratiche e i consigli contenuti in AWS DevOps Well-Architected Guidance e Choosing a Git branching strategy per ambienti multi-account. DevOps Questi ti aiutano a implementare efficacemente lo sviluppo basato su GitFlow, promuovere la collaborazione, migliorare la qualità del codice e semplificare il processo di sviluppo.

Epiche

Revisione dei flussi di lavoro di Gitflow

Operazione	Description	Competenze richieste
Rivedi la procedura standard di Gitflow.	1. Nell'ambiente sandbox, lo sviluppatore crea un `feature` ramo dal `develop` ramo e utilizza lo schema di denominazione. `feature/<ticket>_<initials>_<short description>` 2. Lo sviluppatore sviluppa il codice e lo distribuisce nell'ambiente sandbox in modo iterativo per completare il ticket. **Note** Lo sviluppatore può facoltativamente creare una `sandbox` filiale per eseguire una pipeline di compilazione o distribuzione automatizzata nell'ambiente sandbox. 3. Lo sviluppatore crea una richiesta di unione dal `feature` ramo al ramo	DevOps ingegnere

Operazione	Description	Competenze richieste
	utilizzando uno squash `develop` merge. 4. Una pipeline di integrazione e distribuzione continua (CI/CD) crea e distribuisce automaticamente la filiale nell'ambiente di sviluppo. `develop` 5. (Facoltativo) Uno sviluppatore integra `feature` filiali aggiuntive nel ramo di sviluppo prima di continuare con le attività di rilascio. 6. Quando siete pronti a rilasciare le funzionalità del `develop` ramo, lo sviluppatore crea un `release` ramo denominato in base `release/v<number>` al `develop` ramo. 7. Lo sviluppatore crea il ramo di rilascio, che pubblica artefatti da riutilizzare in altri ambienti. 8. Un approvatore approva manualmente la distribuzione degli artefatti di rilascio nell'ambiente di test. 9. Un approvatore approva manualmente la distribuzione degli elementi di rilascio nell'ambiente di staging.	

Operazione	Description	Competenze richieste
	10. Un approvatore approva manualmente la distribuzione degli elementi di rilascio nell'ambiente di produzione. 11. Lo sviluppatore unisce il ramo al `release` ramo. `main` Idealmente, lo sviluppatore utilizza uno script automatico per eseguire un'unione rapida. Non utilizzare uno squash merge. 12. Lo sviluppatore unisce il `release` ramo al ramo. `develop` Idealmente, lo sviluppatore utilizza uno script automatico per eseguire un'unione rapida. Non utilizzare uno squash merge.	

Operazione	Description	Competenze richieste
Esamina la procedura dell'hotfix Gitflow.	1. Lo sviluppatore crea un `hotfix` ramo dal `main` ramo e utilizza lo schema di denominazione. `hotfix/<ticket>_<initials>_<short description>` 2. Lo sviluppatore crea un `release` ramo dal `main` ramo e gli dà `release/v<number>` un nome. 3. Lo sviluppatore risolve il problema, esegue la correzione e crea il ramo. `hotfix` 4. Lo sviluppatore crea una richiesta di unione dal `hotfix` ramo al ramo utilizzando uno `release/v<number>` squash merge. 5. Lo sviluppatore crea il `release` ramo, che pubblica artefatti da riutilizzare in altri ambienti. 6. Un approvatore approva manualmente la distribuzione degli artefatti di rilascio nell'ambiente di test. 7. Un approvatore approva manualmente la distribuzione degli elementi di rilascio nell'ambiente di staging.	DevOps ingegnere

Operazione	Description	Competenze richieste
	8. Un approvatore approva manualmente la distribuzione degli elementi di rilascio nell'ambiente di produzione. 9. Lo sviluppatore unisce il ramo al `release` ramo. `main` Idealmente, lo sviluppatore utilizza uno script automatico per eseguire un'unione rapida. Non utilizzare uno squash merge. 10. Lo sviluppatore unisce il `release` ramo al ramo. `develop` Idealmente, lo sviluppatore utilizza uno script automatico per eseguire un'unione rapida. Non utilizzare uno squash merge. 11. Se viene rilevato un conflitto, gli sviluppatori ricevono un avviso e risolvono il conflitto con una richiesta di unione.	

Operazione	Description	Competenze richieste
Rivedi il processo di correzione dei bug di Gitflow.	1. Lo sviluppatore crea un `bugfix` ramo dal `release/v<number>` ramo corrente e utilizza il modello di denominazione. `bugfix/<ticket number>_<developer initials>_<descriptor>` 2. Lo sviluppatore risolve il problema, esegue la correzione e crea il ramo. `bugfix` 3. Lo sviluppatore crea una richiesta di unione dal `bugfix` ramo al ramo utilizzando uno `release/v<number>` squash merge. 4. Lo sviluppatore crea il `release` ramo, che pubblica artefatti da riutilizzare in altri ambienti. 5. Un approvatore approva manualmente la distribuzione degli artefatti di rilascio nell'ambiente di test. 6. Un approvatore approva manualmente la distribuzione degli elementi di rilascio nell'ambiente Stage. 7. Un approvatore approva manualmente la distribuzione degli elementi di	DevOps ingegnere

Operazione	Description	Competenze richieste
	rilascio nell'ambiente di produzione. 8. Lo sviluppatore unisce il ramo al `release` ramo. `main` Idealmente, lo sviluppatore utilizza uno script automatico per eseguire un'unione rapida. Non utilizzare uno squash merge. 9. Lo sviluppatore unisce il `release` ramo al ramo. `develop` Idealmente, lo sviluppatore utilizza uno script automatico per eseguire un'unione rapida. Non utilizzare uno squash merge. 10. Se viene rilevato un conflitto, gli sviluppatori ricevono un avviso e risolvono il conflitto con una richiesta di unione.	

risoluzione dei problemi

Problema	Soluzione
conflitti tra filiali	Un problema comune che può verificarsi con il modello Gitflow è quando è necessario un hotfix in produzione, mentre una modifica corrispondente deve avvenire in un ambiente inferiore, dove un'altra filiale sta modificando le stesse risorse. Ti consigliamo di avere un

Problema	Soluzione
	solo ramo di release attivo alla volta. Se ne avete più di uno attivo alla volta, i cambiamenti negli ambienti potrebbero interferire e potreste non essere in grado di portare un ramo alla produzione.
Fusione	I rilasci dovrebbero essere ricongiunti a quelli principali e distribuiti il prima possibile, in modo da concentrare nuovamente il lavoro nelle filiali principali.
Fusione di Squash	Usa uno squash merge solo quando ti unisci da un ramo a un `feature` ramo. `develop` L'uso di unioni a forma di squash nei rami più alti causa difficoltà quando si ricongiungono le modifiche ai rami inferiori.

Risorse correlate

Questa guida non include la formazione per Git; tuttavia, ci sono molte risorse di alta qualità disponibili su Internet se hai bisogno di questa formazione. Ti consigliamo di iniziare dal sito di documentazione di Git.

Le seguenti risorse possono aiutarti nel tuo percorso di ramificazione con Gitflow nel. Cloud AWS

AWS DevOps guida

- AWS DevOps Linee guida
- AWS Architettura di riferimento della pipeline di distribuzione
- Che cos'è DevOps?
- DevOps resources

Guida Gitflow

- Il blog originale di Gitflow (post sul blog di Vincent Driessen)

- Flusso di lavoro Gitflow (Atlassian)
- Gitflow su GitHub: Come usare i flussi di lavoro Git Flow con repository GitHub basati (video) YouTube
- Esempio di avvio di Git Flow (YouTube video)
- Il ramo di rilascio di Gitflow dall'inizio alla fine (video) YouTube

Altre risorse

Metodologia dell'app a dodici fattori (12factor.net)

Implementa una strategia di ramificazione Trunk per ambienti con più account DevOps

Mike Stephens e Rayjan Wilson, Amazon Web Services

Riepilogo

Quando si gestisce un repository di codice sorgente, diverse strategie di ramificazione influiscono sui processi di sviluppo e rilascio del software utilizzati dai team di sviluppo. Esempi di strategie di ramificazione comuni includono Trunk, GitHub Flow e Gitflow. Queste strategie utilizzano rami diversi e le attività svolte in ciascun ambiente sono diverse. Organizations che stanno implementando DevOps processi trarrebbero vantaggio da una guida visiva per aiutarle a comprendere le differenze tra queste strategie di ramificazione. L'utilizzo di questa immagine nell'organizzazione aiuta i team di sviluppo ad allineare il proprio lavoro e a seguire gli standard organizzativi. Questo modello fornisce questa immagine e descrive il processo di implementazione di una strategia di ramificazione Trunk nell'organizzazione.

Questo modello fa parte di una serie di documentazione sulla scelta e l'implementazione di strategie di DevOps ramificazione per organizzazioni con più membri. Account AWS Questa serie è progettata per aiutarti ad applicare la strategia e le migliori pratiche corrette sin dall'inizio, per semplificare la tua esperienza nel cloud. Trunk è solo una possibile strategia di ramificazione che l'organizzazione può utilizzare. Questa serie di documentazione copre anche i modelli di GitHub ramificazione Flow e Gitflow. Se non l'hai già fatto, ti consigliamo di leggere Scelta di una strategia di ramificazione Git per DevOps ambienti multi-account prima di implementare le linee guida di questo modello. Utilizza la due diligence per scegliere la strategia di ramificazione giusta per la tua organizzazione.

Questa guida fornisce un diagramma che mostra come un'organizzazione potrebbe implementare la strategia Trunk. Si consiglia di consultare la AWS DevOps Well-Architected Guidance ufficiale per esaminare le migliori pratiche. Questo modello include attività, passaggi e restrizioni consigliati per ogni fase del DevOps processo.

Prerequisiti e limitazioni

Prerequisiti

- Git, installato. Viene utilizzato come strumento di archiviazione del codice sorgente.
- Draw.io, installato. Questa applicazione viene utilizzata per visualizzare e modificare il diagramma.

Architecture

Architettura Target

Il diagramma seguente può essere usato come un quadrato di Punnett (Wikipedia). Allineate i rami sull'asse verticale con gli AWS ambienti sull'asse orizzontale per determinare quali azioni eseguire in ogni scenario. I numeri indicano la sequenza delle azioni nel flusso di lavoro. In questo esempio si passa da una `feature` filiale all'implementazione in produzione.

Per ulteriori informazioni sugli ambienti e sui Account AWS rami in un approccio Trunk, vedi Scelta di una strategia di ramificazione Git per ambienti con più account DevOps .

Automazione e scalabilità

L'integrazione continua e la distribuzione continua (CI/CD) is the process of automating the software release lifecycle. It automates much or all of the manual processes traditionally required to get new code from an initial commit into production. A CI/CD pipeline encompasses the sandbox, development, testing, staging, and production environments. In each environment, the CI/CD pipeline provisions any infrastructure that is needed to deploy or test the code. By using CI/CD, development teams can make changes to code that are then automatically tested and deployed. CI/CDpipeline) forniscono inoltre governance e barriere ai team di sviluppo, applicando coerenza, standard, best practice e livelli minimi di accettazione per l'accettazione e l'implementazione delle funzionalità. Per ulteriori informazioni, consulta Practicing Continuous Integration and Continuous Delivery su. AWS

AWS offre una suite di servizi per sviluppatori progettati per aiutarti a creare CI/CD pipeline. Ad esempio, AWS CodePipelineè un servizio di distribuzione continua completamente gestito che consente di automatizzare le pipeline di rilascio per aggiornamenti rapidi e affidabili di applicazioni e infrastrutture. AWS CodeBuildcompila il codice sorgente, esegue test e produce ready-to-deploy pacchetti software. Per ulteriori informazioni, consulta Developer Tools on AWS.

Tools (Strumenti)

AWS servizi e strumenti

AWS fornisce una suite di servizi per sviluppatori che è possibile utilizzare per implementare questo modello:

- AWS CodeArtifactè un servizio di repository di artefatti gestito e altamente scalabile che consente di archiviare e condividere pacchetti software per lo sviluppo di applicazioni.

- AWS CodeBuildè un servizio di compilazione completamente gestito che consente di compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS CodeDeployautomatizza le distribuzioni su Amazon Elastic Compute Cloud (Amazon EC2) o istanze AWS Lambda , funzioni o servizi Amazon Elastic Container Service (Amazon ECS) in locale.
- AWS CodePipelineti aiuta a modellare e configurare rapidamente le diverse fasi di un rilascio del software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.

Altri strumenti

- Draw.io Desktop — Un'applicazione per creare diagrammi di flusso e diagrammi.
- Figma è uno strumento di progettazione online progettato per la collaborazione. Il repository di codice contiene modelli in formato.fig per Figma.

Archivio di codice

Questo file sorgente per il diagramma in questo modello è disponibile nel GitHub repository Git Branching Strategy for Trunk. Include file nei formati PNG, draw.io e Figma. È possibile modificare questi diagrammi per supportare i processi dell'organizzazione.

Best practice

Segui le best practice e i consigli contenuti in AWS DevOps Well-Architected Guidance e Choosing a Git branching strategy per ambienti multi-account. DevOps Questi ti aiutano a implementare efficacemente lo sviluppo basato su Trunk, promuovere la collaborazione, migliorare la qualità del codice e semplificare il processo di sviluppo.

Epiche

Revisione del flusso di lavoro Trunk

Operazione	Description	Competenze richieste
Rivedi il processo Trunk standard.	1. Nell'ambiente sandbox, lo sviluppatore crea un `feature` ramo dal	DevOps ingegnere

Operazione	Description	Competenze richieste
	`main` ramo e utilizza lo schema di denominazione. `feature/<ticket>_<initials>_<short description>` 2. Lo sviluppatore sviluppa il codice e lo distribuisce nell'ambiente sandbox in modo iterativo per completare il ticket. **Note** Lo sviluppatore può facoltativamente creare una `sandbox` filiale per eseguire una pipeline di compilazione o distribuzione automatizzata nell'ambiente sandbox. 3. Lo sviluppatore crea una richiesta di unione dal `feature` ramo al ramo utilizzando uno squash `main` merge. 4. Una pipeline di integrazione e distribuzione continua (CI/CD) crea e pubblica automaticamente gli artefatti dalla filiale all'ambiente di sviluppo. `main`	

Operazione	Description	Competenze richieste
	5. Un approvatore approva manualmente la distribuz ione degli artefatti di rilascio nell'ambiente di sviluppo. 6. Un approvatore approva manualmente la distribuz ione degli elementi di rilascio nell'ambiente di test. 7. Un approvatore approva manualmente la distribuz ione degli elementi di rilascio nell'ambiente di staging. 8. Un approvatore approva manualmente la distribuz ione degli elementi di rilascio nell'ambiente di produzione.	

risoluzione dei problemi

Problema	Soluzione
conflitti tra filiali	Un problema comune che può verificarsi con il modello Trunk è rappresentato dalla necessità di applicare un hotfix in produzione, mentre una modifica corrispondente deve avvenire in una `feature` filiale, dove vengono modificate le stesse risorse. Si consiglia di unire frequente mente le modifiche provenienti dai `main` rami inferiori per evitare conflitti significativi durante l'unione a. `main`

Risorse correlate

Questa guida non include la formazione per Git; tuttavia, ci sono molte risorse di alta qualità disponibili su Internet se hai bisogno di questa formazione. Ti consigliamo di iniziare dal sito di documentazione di Git.

Le seguenti risorse possono aiutarti nel tuo percorso di ramificazione di Trunk in. Cloud AWS

AWS DevOps guida

- AWS DevOps Linee guida
- AWS Deployment Pipeline Reference Architecture
- Che cos'è DevOps?
- DevOps resources

Guida al bagagliaio

- Sviluppo basato sul tronco

Altre risorse

- Metodologia delle app a dodici fattori (12factor.net)

Implementa la scansione Checkov centralizzata e personalizzata per applicare le policy prima di implementare l'infrastruttura AWS

Benjamin Morris, Amazon Web Services

Riepilogo

Questo modello fornisce un framework GitHub Actions per scrivere politiche Checkov personalizzate in un unico repository che può essere riutilizzato in un'organizzazione. GitHub Seguendo questo schema, un team di sicurezza delle informazioni può scrivere, aggiungere e mantenere politiche personalizzate in base ai requisiti aziendali. Le politiche personalizzate possono essere inserite automaticamente in tutte le pipeline dell' GitHub organizzazione. Questo approccio può essere utilizzato per applicare gli standard aziendali relativi alle risorse prima che le risorse vengano impiegate.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un' GitHub organizzazione che utilizza GitHub Actions
- AWS infrastruttura implementata con HashiCorp Terraform o AWS CloudFormation

Limitazioni

- Questo modello è stato scritto per GitHub Actions. Tuttavia, può essere adattato a framework simili di integrazione continua e distribuzione continua (CI/CD) come. GitLab Non è richiesta alcuna versione specifica a pagamento di. GitHub
- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per informazioni sulla disponibilità regionale, consulta Endpoint e quote del servizio nella AWS documentazione e scegli il link relativo al servizio.

Architecture

Questo modello è progettato per essere distribuito come GitHub archivio contenente un flusso di lavoro GitHub riutilizzabile e politiche Checkov personalizzate. Il flusso di lavoro riutilizzabile può scansionare sia i repository Terraform che quelli CloudFormation Infrastructure as Code (IaC).

Il diagramma seguente mostra il repository dei GitHub flussi di lavoro riutilizzabili e l'archivio delle politiche di Checkov personalizzate come icone separate. Tuttavia, è possibile implementare questi repository come repository separati o come repository singolo. Il codice di esempio utilizza un unico repository, con file per i flussi di lavoro (`.github/workflows`) e file per le politiche personalizzate (`custom_policies`cartella e file di `.checkov.yml` configurazione) nello stesso repository.

Il diagramma mostra il flusso di lavoro seguente:

1. Un utente crea una pull request in un repository. GitHub
2. I flussi di lavoro della pipeline iniziano in GitHub Actions, incluso un riferimento a un flusso di lavoro riutilizzabile di Checkov.
3. Il flusso di lavoro della pipeline scarica il flusso di lavoro riutilizzabile Checkov di riferimento da un archivio esterno ed esegue tale flusso di lavoro Checkov utilizzando Actions. GitHub
4. Il flusso di lavoro riutilizzabile di Checkov scarica le politiche personalizzate da un archivio esterno.
5. Il flusso di lavoro riutilizzabile di Checkov valuta l'IaC nel GitHub repository rispetto alle politiche Checkov integrate e personalizzate. Il flusso di lavoro riutilizzabile di Checkov ha esito positivo o negativo a seconda che vengano rilevati problemi di sicurezza.

Automazione e scalabilità

Questo modello consente la gestione centrale della configurazione di Checkov, in modo che gli aggiornamenti delle policy possano essere applicati in un'unica posizione. Tuttavia, questo modello richiede che ogni repository utilizzi un flusso di lavoro che contenga un riferimento al flusso di lavoro riutilizzabile centrale. È possibile aggiungere questo riferimento manualmente o utilizzare script per inviare il file alla `.github/workflows` cartella di ciascun repository.

Tools (Strumenti)

Servizi AWS

- CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita in tutte le regioni. Account AWS Checkov può eseguire la scansione. CloudFormation

Altri strumenti

- Checkov è uno strumento di analisi statica del codice che verifica la presenza di configurazioni errate di sicurezza e conformità in IaC.
- GitHub Actions è integrato nella GitHub piattaforma per aiutarti a creare, condividere ed eseguire flussi di lavoro all'interno dei tuoi repository. GitHub Puoi utilizzare GitHub Actions per automatizzare attività come la creazione, il test e la distribuzione del codice.
- Terraform è uno strumento IaC HashiCorp che ti aiuta a creare e gestire risorse cloud e locali. Checkov può scansionare Terraform.

Archivio di codici

Il codice per questo pattern è disponibile nel GitHub centralized-custom-checkov-sastrepository.

Best practice

- Per mantenere un livello di sicurezza coerente, allinea le politiche di sicurezza della tua azienda alle politiche di Checkov.
- Nelle prime fasi di implementazione delle politiche personalizzate di Checkov, è possibile utilizzare l'opzione soft-fail nella scansione di Checkov per consentire l'unione delle IAC con problemi di sicurezza. Man mano che il processo matura, passa dall'opzione soft-fail all'opzione hard-fail.

Epiche

Crea un repository Checkov centrale per le politiche personalizzate

Operazione	Description	Competenze richieste
Crea un repository centrale di Checkov.	Crea un repository per archiviare le politiche di Checkov personalizzate che verranno utilizzate all'interno dell'organizzazione. Per iniziare rapidamente, puoi copiare il contenuto dell'arch ivio di questo pattern nel tuo GitHub centralized-custom	DevOps ingegnere

Operazione	Description	Competenze richieste
	-checkov-sast repository centrale di Checkov.	
Crea un archivio per flussi di lavoro riutilizzabili.	Se esiste già un archivio per flussi di lavoro riutilizz abili o prevedi di includere file di workflow riutilizzabili nello stesso repository delle politiche personalizzate di Checkov, puoi saltare questo passaggio. Crea un repository per contenere flussi di lavoro riutilizzabili. GitHub Le pipeline di altri repository faranno riferimento a questo repositor y.	DevOps ingegnere

Crea flussi di lavoro Checkov riutilizzabili ed esempi

Operazione	Description	Competenze richieste
Aggiungi un flusso di lavoro Checkov riutilizzabile.	Crea un flusso di lavoro Checkov GitHub Actions riutilizzabile (file YAML) nel repository dei flussi di lavoro riutilizzabili. È possibile adattare questo flusso di lavoro riutilizzabile dal file di flusso di lavoro fornito in questo modello. Un esempio di modifica che potreste voler apportare è	DevOps ingegnere

Operazione	Description	Competenze richieste
	modificare il flusso di lavoro riutilizzabile utilizzando l'opzione soft-fail. L'impostazione `soft-fail` su `true` consente il completamento corretto del processo anche in caso di scansione Checkov non riuscita. Per istruzioni, consulta Hard and soft fail nella documentazione di Checkov.	
Aggiungi un esempio di workflow.	Aggiungi un esempio di flusso di lavoro Checkov che fa riferimento al `reusable` flusso di lavoro. Ciò fornirà un modello su come riutilizzare il `reusable` flusso di lavoro. Nel repository di esempio, `checkov-source.yaml` è il flusso di lavoro riutilizzabile ed `checkov-scan.yaml` è l'esempio che consuma. `checkov-source` Per maggiori dettagli sulla scrittura di un esempio di workflow Checkov, consulta Informazioni aggiuntive.	DevOps ingegnere

Associa le politiche aziendali alle politiche personalizzate di Checkov

Operazione	Description	Competenze richieste
Determina le politiche che possono essere applicate con Checkov.	1. Esamina le politiche aziendali relative alla sicurezza dell'infrastruttura e quali requisiti devono essere rispettati. 2. Determina quali requisiti possono essere implementati utilizzando le politiche personalizzate di Checkov. 3. Crea una convenzione di denominazione che associ il controllo delle politiche alla politica personalizzata di Checkov. In genere, le politiche personalizzate di Checkov hanno un identificatore con un nome Checkov, l'origine della politica (personalizzata) e un numero di politica (ad esempio,). `CKV2_CUSTOM_123` Per maggiori dettagli sulla creazione di politiche personalizzate di Checkov, vedere Panoramica delle politiche personalizzate nella documentazione di Checkov.	Sicurezza e conformità
Aggiungi le politiche personalizzate di Checkov.	Converti le politiche aziendali identificate in politiche	Sicurezza

Operazione	Description	Competenze richieste
	Checkov personalizzate nell'archivio centrale. Puoi scrivere semplici politiche di Checkov in Python o YAML.	

Implementa politiche personalizzate centralizzate di Checkov

Operazione	Description	Competenze richieste
Aggiungi il flusso di lavoro riutilizzabile di Checkov a tutti i repository.	A questo punto, dovresti avere un esempio di flusso di lavoro Checkov che faccia riferimento al flusso di lavoro riutilizzabile. Copia il flusso di lavoro Checkov di esempio che fa riferimento al flusso di lavoro riutilizzabile in ogni repository che lo richiede.	DevOps ingegnere
Crea un meccanismo per garantire che Checkov venga eseguito prima delle fusioni.	Per garantire che il flusso di lavoro Checkov venga eseguito per ogni richiesta pull, crea un controllo dello stato che richieda un flusso di lavoro Checkov corretto prima di poter unire le richieste pull. GitHub consente di richiedere l'esecuzione di flussi di lavoro specifici prima di poter unire le pull request.	DevOps ingegnere
Crea un PAT a livello di organizzazione e condividilo come segreto.	Se la tua GitHub organizzazione è visibile pubblicamente, puoi saltare questo passaggio.	DevOps ingegnere

Operazione	Description	Competenze richieste
	Questo modello richiede che il flusso di lavoro di Checkov sia in grado di scaricare le politiche personalizzate dall'archivio delle politiche personalizzate dell'organizzazione. GitHub È necessario fornire le autorizzazioni necessarie per consentire al flusso di lavoro di Checkov di accedere a tali archivi. A tale scopo, create un token di accesso personale (PAT) con le autorizzazioni per leggere gli archivi dell'organizzazione. Condividi questo PAT con gli archivi, come segreto a livello di organizzazione (se hai un piano a pagamento) o come segreto in ogni repository (versione gratuita). Nel codice di esempio, il nome predefinito per il segreto è. `ORG_PAT`	

Operazione	Description	Competenze richieste
(Facoltativo) Proteggi i file del flusso di lavoro Checkov dalle modifiche.	Per proteggere i file del flusso di lavoro Checkov da modifiche indesider ate, è possibile utilizzare un `CODEOWNERS` file. Il `CODEOWNERS` file viene in genere distribuito nella radice della directory. Ad esempio, per richiedere l'approvazione del `secEng` gruppo GitHub dell'orga nizzazione quando il `checkov-scan.yaml` file viene modificato, aggiungi quanto segue al file di un repository: `CODEOWNERS` `[Checkov] .github/workflows /checkov-scan.yaml @myOrg/secEng` Un `CODEOWNERS` file è specifico del repository in cui risiede. Per proteggere il flusso di lavoro Checkov utilizzato dal repository, è necessario aggiungere (o aggiornare) un `CODEOWNERS` file in ogni repository. Per ulteriori informazioni sulla protezione dei file del flusso di lavoro di Checkov, consulta Informazioni aggiuntive.	DevOps ingegnere

Operazione	Description	Competenze richieste
	Per ulteriori informazioni sui `CODEOWNERS` file, consultate la documentazione ufficiale del CI/CD provider (ad esempio GitHub).	

Risorse correlate

- Panoramica delle politiche personalizzate di Checkov
- CloudFormation Scansione della configurazione
- GitHub Azioni Flussi di lavoro riutilizzabili

Informazioni aggiuntive

Scrittura di file di workflow Checkov

Quando scrivi`checkov-scan.yaml`, considera quando vuoi che venga eseguito. La on chiave di primo livello determina quando viene eseguito il flusso di lavoro. Nel repository di esempio, il flusso di lavoro verrà eseguito quando è presente una richiesta pull destinata al `main` ramo (e ogni volta che il ramo sorgente della pull request viene modificato). Il flusso di lavoro può anche essere eseguito come richiesto grazie alla `workflow_dispatch` chiave.

È possibile modificare le condizioni di attivazione del flusso di lavoro in base alla frequenza con cui si desidera che venga eseguito il flusso di lavoro. Ad esempio, è possibile modificare il flusso di lavoro in modo che venga eseguito ogni volta che il codice viene inviato a qualsiasi ramo `pull_request` sostituendo `push` e rimuovendo la `branches` chiave.

È possibile modificare il file di workflow di esempio creato all'interno di un singolo repository. Ad esempio, è possibile modificare il nome del ramo di destinazione da `main` a `production` se un repository è strutturato attorno a un `production` ramo.

Protezione dei file del flusso di lavoro Checkov

La scansione Checkov fornisce informazioni utili su potenziali errori di configurazione della sicurezza. Tuttavia, alcuni sviluppatori potrebbero percepirla come un ostacolo alla loro produttività e tentare di rimuovere o disabilitare il flusso di lavoro di scansione.

Esistono diversi modi per risolvere questo problema, tra cui una migliore messaggistica sul valore a lungo termine della scansione di sicurezza e una documentazione più chiara su come implementare un'infrastruttura sicura. Si tratta di importanti approcci «morbidi» alla DevSecOps collaborazione che possono essere visti come la soluzione alla causa principale di questo problema. Tuttavia, puoi anche utilizzare controlli tecnici come un `CODEOWNERS` file come guardrail per aiutare gli sviluppatori a seguire la strada giusta.

Modello di test in una sandbox

Per testare questo pattern in un ambiente sandbox, segui questi passaggi:

1. Crea una nuova GitHub organizzazione. Crea un token con accesso in sola lettura a tutti gli archivi dell'organizzazione. Poiché questo token è destinato a un ambiente sandbox, non a un ambiente a pagamento, non sarà possibile archiviarlo in un segreto a livello di organizzazione.
2. Crea un `checkov` repository per contenere la configurazione di Checkov e un `github-workflows` repository per contenere la configurazione del flusso di lavoro riutilizzabile. Compila i repository con il contenuto del repository di esempio.
3. Create un repository di applicazioni e copiate e incollate il `checkov-scan.yaml` flusso di lavoro nella relativa cartella. `.github/workflows` Aggiungi un segreto al repository che contiene il PAT che hai creato per l'accesso in sola lettura all'organizzazione. Il segreto predefinito è. `ORG_PAT`
4. Crea una pull request che aggiunga Terraform o CloudFormation codice al repository dell'applicazione. Checkov dovrebbe scansionare e restituire un risultato.

Implementa la diagnostica e la risoluzione dei problemi di Kubernetes basate sull'intelligenza artificiale con K8SGPT e l'integrazione con Amazon Bedrock

Ishwar Chauthaiwale, Muskan., e Prafful Gupta, Amazon Web Services

Riepilogo

Questo modello dimostra come implementare la diagnostica e la risoluzione dei problemi di Kubernetes basate sull'intelligenza artificiale integrando K8SGPT con il modello Anthropic Claude v2 disponibile su Amazon Bedrock. La soluzione fornisce analisi in linguaggio naturale e procedure di correzione per i problemi del cluster Kubernetes attraverso un'architettura host bastion sicura. Combinando l'esperienza di K8SGPT Kubernetes con le funzionalità linguistiche avanzate di Amazon Bedrock DevOps , i team possono identificare e risolvere rapidamente i problemi dei cluster. Con queste funzionalità, è possibile ridurre il tempo medio di risoluzione (MTTR) fino al 50%.

Questo modello nativo del cloud sfrutta Amazon Elastic Kubernetes Service (Amazon EKS) per la gestione di Kubernetes. Il modello implementa le migliori pratiche di sicurezza attraverso ruoli (IAM) appropriati e isolamento della rete. AWS Identity and Access Management Questa soluzione è particolarmente utile per le organizzazioni che desiderano semplificare le operazioni Kubernetes e migliorare le capacità di risoluzione dei problemi con l'assistenza dell'IA.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo con le autorizzazioni appropriate Account AWS
- AWS Command Line Interface (AWS CLI) installato e configurato
- Cluster Amazon EKS
- Accesso al modello Anthropic Claude 2 su Amazon Bedrock
- Un host bastion con le impostazioni richieste per i gruppi di sicurezza
- K8sgpt installato

Limitazioni

- L'analisi K8sgpt è limitata dalla dimensione della finestra di contesto del modello Claude v2.

- I limiti di tariffa dell'API Amazon Bedrock si applicano in base alle quote del tuo account.
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità regionale, vedi AWS Servizi per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- Amazon EKS versione 1.31 o successiva
- Modello Claude 2 su Amazon Bedrock
- K8SGpt v0.4.2 o versione successiva

Architecture

Il diagramma seguente mostra l'architettura per la diagnostica Kubernetes basata sull'intelligenza artificiale che utilizza K8sgPT integrato con Amazon Bedrock in. Cloud AWS

L'architettura mostra il seguente flusso di lavoro:

1. Gli sviluppatori accedono all'ambiente tramite una connessione sicura all'host bastion. Questa EC2 istanza Amazon funge da punto di ingresso sicuro e contiene l'installazione dell'interfaccia a riga di comando (CLI) K8SGpt e le configurazioni richieste.
2. L'host bastion, configurato con ruoli IAM specifici, stabilisce connessioni sicure sia al cluster Amazon EKS che agli endpoint Amazon Bedrock. K8sgpt è installato e configurato sull'host bastion per eseguire l'analisi del cluster Kubernetes.
3. Amazon EKS gestisce il piano di controllo Kubernetes e i nodi di lavoro, fornendo l'ambiente di destinazione per l'analisi K8sgPT. Il servizio funziona su più zone di disponibilità all'interno di un cloud privato virtuale (VPC), il che aiuta a fornire disponibilità e resilienza elevate. Amazon EKS fornisce dati operativi tramite l'API Kubernetes, che consente un'analisi completa dei cluster.
4. K8sgPT invia i dati di analisi ad Amazon Bedrock, che fornisce il Claude v2 Foundation Model (FM) per l'elaborazione del linguaggio naturale. Il servizio elabora l'analisi K8sgpt per generare spiegazioni leggibili dall'uomo e offre suggerimenti di correzione dettagliati in base ai problemi identificati. Amazon Bedrock funziona come un servizio di intelligenza artificiale senza server con disponibilità e scalabilità elevate.

> **Note**
>
> In tutto questo flusso di lavoro, IAM controlla l'accesso tra i componenti tramite ruoli e policy, gestendo l'autenticazione per le interazioni tra bastion host, Amazon EKS e Amazon Bedrock. IAM implementa il principio del privilegio minimo e consente una comunicazione sicura tra servizi in tutta l'architettura.

Automazione e scalabilità

Le operazioni K8sgPT possono essere automatizzate e scalate su più cluster Amazon EKS tramite vari strumenti. Servizi AWS Questa soluzione supporta l'integrazione continua e l'integrazione della distribuzione continua (CI/CD) utilizzando Jenkins, Actions o per l'analisi pianificata. GitHub AWS CodeBuild L'operatore K8SGPT consente il monitoraggio continuo all'interno del cluster con funzionalità automatiche di rilevamento e segnalazione dei problemi. Per le implementazioni su scala aziendale, puoi utilizzare Amazon EventBridge per pianificare scansioni e attivare risposte automatiche con script personalizzati. AWS L'integrazione SDK consente il controllo programmatico su un'ampia flotta di cluster.

Tools (Strumenti)

Servizi AWS

- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando.
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire AWS Kubernetes senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.

Altri strumenti

- K8SGPT è uno strumento open source basato sull'intelligenza artificiale che trasforma la gestione di Kubernetes. Agisce come esperto di Virtual Site Reliability Engineering (SRE), scansionando, diagnosticando e risolvendo automaticamente i problemi del cluster Kubernetes. Gli amministratori possono interagire con K8SGPT utilizzando il linguaggio naturale e ottenere informazioni chiare e fruibili sullo stato del cluster, sugli arresti anomali dei pod e sugli errori del servizio. Gli analizzatori

integrati dello strumento rilevano un'ampia gamma di problemi, dai componenti mal configurati ai vincoli di risorse, e forniscono spiegazioni e soluzioni. easy-to-understand

Best practice

- Implementa controlli di accesso sicuri utilizzando for bastion host access. AWS Systems Manager Session Manager
- Assicurati che l'autenticazione K8sgpt utilizzi ruoli IAM dedicati con autorizzazioni con privilegi minimi per le interazioni tra Amazon Bedrock e Amazon EKS. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.
- Configura il tagging delle risorse, abilita la CloudWatch registrazione di Amazon per gli audit trail e implementa l'anonimizzazione dei dati per le informazioni sensibili.
- Mantieni backup regolari delle configurazioni K8SGPT mentre imposta pianificazioni di scansione automatizzate durante le ore non di punta per ridurre al minimo l'impatto operativo.

Epiche

Aggiungi Amazon Bedrock all'elenco dei provider di backend AI.

Operazione	Description	Competenze richieste
Imposta Amazon Bedrock come provider di backend AI per K8sgPT.	Per impostare Amazon Bedrock come fornitore di backend AI r per K8sgpt, usa il seguente comando: AWS CLI `k8sgpt auth add -b amazonbedrock \ -r us-west-2 \ -m anthropic.claude-v2 \ -n endpoint-name` Il comando di esempio utilizza per. `us-west-2` Regione AWS Tuttavia, puoi seleziona	AWS DevOps

Operazione	Description	Competenze richieste

	re un'altra regione, a condizion e che sia il cluster Amazon EKS che il modello Amazon Bedrock corrispondente siano disponibili e abilitati nella regione selezionata.
	
	Per verificare che `amazonbed rock` sia stato aggiunto all'elenco dei provider di backend AI e sia nello `Active` stato, esegui il comando seguente:
	
	`k8sgpt auth list`
	
	Di seguito è riportato un esempio dell'output previsto di questo comando:
	
	`Default:`
	`> openai`
	`Active:`
	`> amazonbedrock`
	`Unused:`
	`> openai`
	`> localai`
	`> ollama`
	`> azureopenai`
	`> cohere`
	`> amazonsagemaker`
	`> google`
	`> noopai`
	`> huggingface`
	`> googlevertexai`
	`> oci`
	`> customrest`	

Operazione	Description	Competenze richieste
	`> ibmwatsonxai`	

Scansiona le risorse utilizzando un filtro

Operazione	Description	Competenze richieste

Visualizza un elenco di filtri disponibili.	Per visualizzare l'elenco di tutti i filtri disponibili, usa il seguente AWS CLI comando:
	`k8sgpt filters list`
	Di seguito è riportato un esempio dell'output previsto di questo comando:
	`Active:`
	`> Deployment`
	`> ReplicaSet`
	`> PersistentVolumeClaim`
	`> Service`
	`> CronJob`
	`> Node`
	`> MutatingWebhookConfiguration`
	`> Pod`
	`> Ingress`
	`> StatefulSet`
	`> ValidatingWebhookConfiguration`
		AWS DevOps

Scansiona un pod in uno spazio dei nomi specifico utilizzando un filtro.	Questo comando è utile per il debug mirato di problemi specifici dei pod all'interno di un cluster Kubernetes,	AWS DevOps

Operazione	Description	Competenze richieste

	utilizzando le funzionalità AI di Amazon Bedrock per analizzare e spiegare i problemi rilevati.
	
	Per scansionare un pod in uno spazio dei nomi specifico utilizzando un filtro, usa il seguente comando: AWS CLI
	
	`k8sgpt analyze --backend amazonbedrock --explain --filter Pod -n default`
	
	Di seguito è riportato un esempio dell'output previsto di questo comando:
	
	`100% |################################################| (1/1, 645 it/s)`
	
	`AI Provider: amazonbedrock`
	
	`0: Pod default/crashme()`
	`- Error: the last termination reason is Error container=crashme pod=crashme`
	`Error: The pod named crashme terminated because the container named crashme crashed.`
	`Solution: Check logs for crashme pod to identify reason for`	

Operazione	Description	Competenze richieste
	`crash. Restart pod or redeploy application to resolve crash.`	

Operazione	Description	Competenze richieste

Scansiona una distribuzione in uno spazio dei nomi specifico utilizzando un filtro.	Questo comando è utile per identificare e risolvere problemi specifici della distribuzione, in particola re quando lo stato effettivo non corrisponde allo stato desiderato.
	
	Per eseguire la scansione di una distribuzione in uno spazio dei nomi specifico utilizzando un filtro, utilizzare il comando seguente: AWS CLI
	
	`k8sgpt analyze -- backend amazonbedrock --explain --filter Deployment -n default`
	
	Di seguito è riportato un esempio dell'output previsto di questo comando:
	
	`100% |################## ################## ################## #####| (1/1, 10 it/min)`
	
	`AI Provider: amazonbed rock`
	
	`0: Deployment default/n ginx()`
	`- Error: Deployment default/nginx has 1 replicas but 2 are available`
		AWS DevOps

Operazione	Description	Competenze richieste
	`Error: The Deployment named nginx in the default namespace has 1 replica specified but 2 pod replicas are running.` `Solution: Check if any other controllers like ReplicaSet or StatefulSet have created extra pods. Delete extra pods or adjust replica count to match available pods.`	

Scansiona un nodo in uno spazio dei nomi specifico utilizzando un filtro.	Per eseguire la scansione di un nodo in uno spazio dei nomi specifico utilizzando un filtro, utilizzate il seguente comando: AWS CLI
	`k8sgpt analyze --backend amazonbedrock --explain --filter Node -n default`
	Di seguito è riportato un esempio dell'output previsto di questo comando:
	`AI Provider: amazonbedrock`
	
	`No problems detected`
		AWS DevOps

Analizza gli output dettagliati

Operazione	Description	Competenze richieste

Ottieni risultati dettagliati.	Per ottenere risultati dettagliati, usa il seguente comando: AWS CLI
	`k8sgpt analyze --backend amazonbedrock --explain --ouput json`
	Di seguito è riportato un esempio dell'output previsto di questo comando:
	`{ "provider": "amazonbedrock", "errors": null, "status": "ProblemDetected", "problems": 1, "results": [ { "kind": "Pod", "name": "default/crashme", "error": [ { "Text": "the last termination reason is Error container=crashme pod=crashme", "Kubernetes Doc": "", "Sensitive": [] } ],`
		AWS DevOps

Operazione	Description	Competenze richieste
	"details": " Error: The pod named crashme terminated because the container named crashme crashed. \nSolution: Check logs for crashme pod to identify reason for crash. Restart pod or redeploy application to resolve crash.", "parentObject": "" }] }	

Controlla i pod problematici.	Per verificare la presenza di pod problematici specifici, utilizzate il seguente comando: AWS CLI
	`kubectl get pods --all-namespaces | grep -v Running`
	Di seguito è riportato un esempio dell'output previsto di questo comando:
	`NAMESPACE NAME READY STATUS RESTARTS AGE`
	`default crashme 0/1 CrashLoopBackOff 260(91s ago) 21h`
		AWS DevOps

Operazione	Description	Competenze richieste

Ottieni approfondimenti specifici per le applicazioni.	Questo comando è particola rmente utile quando:
	• Vuoi concentrarti su un'applicazione specifica del tuo cluster.
	• Utilizzi le etichette in modo efficace per organizzare le tue risorse Kubernetes.
	• È necessario verificare rapidamente lo stato di un particolare componente dell'applicazione.
	
	Per ottenere informazioni specifiche sull'applicazione, utilizzate il seguente comando:
	`k8sgpt analyze --backend amazonbedrock --explain -L app=nginx -n default`
	
	Di seguito è riportato un esempio dell'output previsto di questo comando:
	`AI Provider: amazonbedrock`
	
	`No problems detected`	

Risorse correlate

Blog AWS

- Automatizza la risoluzione dei problemi di Amazon EKS utilizzando un flusso di lavoro agentico Amazon Bedrock
- Usa K8sgPT e Amazon Bedrock per la manutenzione semplificata del cluster Kubernetes

AWS documentazione

- AWS CLI comandi: create-cluster e describe-cluster
- Inizia a usare Amazon EKS (documentazione Amazon EKS)
- Best practice di sicurezza in IAM (documentazione IAM)

Altre risorse

- K8sGpt

Rileva automaticamente le modifiche e avvia diverse CodePipeline pipeline per un monorepo in CodeCommit

Helton Ribeiro, Petrus Batalha e Ricardo Morais, Amazon Web Services

Riepilogo

Avviso: AWS Cloud9 non è più disponibile per i nuovi clienti. I clienti esistenti di AWS Cloud9 possono continuare a utilizzare il servizio normalmente. Ulteriori informazioni

Questo modello consente di rilevare automaticamente le modifiche al codice sorgente di un'applicazione basata su monorepo AWS CodeCommit e quindi di avviare una pipeline AWS CodePipeline che esegua l'integrazione continua e la distribuzione continua (CI/CD) automation for each microservice. This approach means that each microservice in your monorepo-based application can have a dedicated CI/CDpipeline), che garantisce una migliore visibilità, una condivisione più semplice del codice e una migliore collaborazione, standardizzazione e reperibilità.

La soluzione descritta in questo modello non esegue alcuna analisi delle dipendenze tra i microservizi all'interno del monorepo. Rileva solo le modifiche nel codice sorgente e avvia la pipeline corrispondente. CI/CD

Il modello viene utilizzato AWS Cloud9 come ambiente di sviluppo integrato (IDE) e AWS Cloud Development Kit (AWS CDK) per definire un'infrastruttura utilizzando due CloudFormation stack: e. `MonoRepoStack` `PipelinesStack` Lo `MonoRepoStack` stack crea il monorepo in AWS CodeCommit e la AWS Lambda funzione che avvia le pipeline. CI/CD Lo `PipelinesStack` stack definisce l'infrastruttura della pipeline.

> **Important**
>
> Il flusso di lavoro di questo pattern è un proof of concept (PoC). Si consiglia di utilizzarlo solo in un ambiente di test. Se desideri utilizzare l'approccio di questo modello in un ambiente di produzione, consulta le migliori pratiche di sicurezza in IAM nella documentazione AWS Identity and Access Management (IAM) e apporta le modifiche necessarie ai tuoi ruoli IAM e Servizi AWS.

Prerequisiti e limitazioni

Prerequisiti

- Un AWS account attivo.
- AWS Command Line Interface (AWS CLI), installato e configurato. Per ulteriori informazioni, vedere Installazione, aggiornamento e disinstallazione di AWS CLI nella AWS CLI documentazione.
- Python 3 `epip`, installato sul computer locale. Per ulteriori informazioni, consulta la documentazione di Python.
- AWS CDK, installato e configurato. Per ulteriori informazioni, vedere Guida introduttiva AWS CDK a nella AWS CDK documentazione.
- Un AWS Cloud9 IDE, installato e configurato. Per ulteriori informazioni, consulta Configurazione AWS Cloud9 nella AWS Cloud9 documentazione.
- Il repository GitHub AWS CodeCommit monorepo multi-pipeline Triggers, clonato sul computer locale.
- Una directory esistente contenente il codice dell'applicazione che si desidera creare e utilizzare. CodePipeline
- Familiarità ed esperienza con le DevOps migliori pratiche su. Cloud AWS Per aumentare la tua familiarità con DevOps, puoi utilizzare il modello Costruisci un'architettura ad accoppiamento libero con microservizi utilizzando DevOps pratiche e AWS Cloud9 sul sito Web Prescriptive Guidance. AWS

Architecture

Il diagramma seguente mostra come utilizzare per definire un'infrastruttura con due AWS CDK stack: e. AWS CloudFormation `MonoRepoStack` `PipelinesStack`

Il diagramma mostra il flusso di lavoro seguente:

1. Il processo di bootstrap utilizza AWS CDK per creare gli stack e. AWS CloudFormation `MonoRepoStack` `PipelinesStack`
2. Lo `MonoRepoStack` stack crea il CodeCommit repository per l'applicazione e la funzione `monorepo-event-handler` Lambda che viene avviata dopo ogni commit.
3. Lo `PipelinesStack` stack crea le pipeline CodePipeline avviate dalla funzione Lambda. Ogni microservizio deve avere una pipeline di infrastruttura definita.
4. La pipeline for `microservice-n` viene avviata dalla funzione Lambda e inizia le sue CI/CD fasi isolate basate sul codice sorgente in. CodeCommit

5. La pipeline for `microservice-1` viene avviata dalla funzione Lambda e inizia le sue CI/CD fasi isolate basate sul codice sorgente in. CodeCommit

Il diagramma seguente mostra la distribuzione degli AWS CloudFormation stack `MonoRepoStack` e in un account. `PipelinesStack`

1. Un utente modifica il codice in uno dei microservizi dell'applicazione.
2. L'utente invia le modifiche da un repository locale a un repository. CodeCommit
3. L'attività push avvia la funzione Lambda che riceve tutti i push al repository. CodeCommit
4. La funzione Lambda legge un parametro in Parameter Store, una funzionalità di AWS Systems Manager, per recuperare l'ID di commit più recente. Il parametro ha il formato di denominazione:. `/MonoRepoTrigger/{repository}/{branch_name}/LastCommit` Se il parametro non viene trovato, la funzione Lambda legge l'ultimo ID di commit dal CodeCommit repository e salva il valore restituito in Parameter Store.
5. Dopo aver identificato l'ID di commit e i file modificati, la funzione Lambda identifica le pipeline per ogni directory di microservizi e avvia la pipeline richiesta. CodePipeline

Tools (Strumenti)

- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software per definire l'infrastruttura cloud in codice e fornirla tramite. CloudFormation
- Python è un linguaggio di programmazione che consente di lavorare rapidamente e integrare i sistemi in modo più efficace.

Codice

Il codice sorgente e i modelli per questo pattern sono disponibili nel repository GitHub AWS CodeCommit monorepo multi-pipeline triggers.

Best practice

- Questa architettura di esempio non include una soluzione di monitoraggio per l'infrastruttura implementata. Se desideri implementare questa soluzione in un ambiente di produzione, ti consigliamo di abilitare il monitoraggio. Per ulteriori informazioni, consulta Monitoraggio delle

applicazioni serverless con CloudWatch Application Insights nella documentazione AWS Serverless Application Model (AWS SAM).

- Quando modifichi il codice di esempio fornito da questo modello, segui le migliori pratiche per lo sviluppo e la distribuzione dell'infrastruttura cloud riportate nella AWS CDK documentazione.
- Quando definisci le tue pipeline di microservizi, consulta le migliori pratiche di sicurezza nella documentazione. AWS CodePipeline
- Puoi anche verificare le migliori pratiche nel AWS CDK codice utilizzando l'utilità cdk-nag. Questo strumento utilizza una serie di regole, raggruppate per pacchetti, per valutare il codice. I pacchetti disponibili sono:
  - AWS Libreria di soluzioni
  - Sicurezza dell'Health Insurance Portability and Accountability Act (HIPAA)
  - Istituto nazionale di standard e tecnologia (NIST) 800-53 rev 4
  - NIST 800-53 rev 5
  - Standard di sicurezza dei dati del settore delle carte di pagamento (PCI DSS) 3.2.1

Epiche

Configura l'ambiente

Operazione	Description	Competenze richieste
Crea un ambiente Python virtuale.	Nel tuo AWS Cloud9 IDE, crea un ambiente Python virtuale e installa le dipendenz e richieste eseguendo il seguente comando: `make install`	Developer
Avvia il comando Account AWS e per. Regione AWS AWS CDK	Esegui il bootstrap di required Account AWS e Region eseguendo il seguente comando: `make bootstrap account-id=<your-`	Developer

Operazione	Description	Competenze richieste
	`AWS-account-ID> region=<required-region>`	

Aggiungi una nuova pipeline per un microservizio

Operazione	Description	Competenze richieste
Aggiungete il codice di esempio alla directory dell'applicazione.	Aggiungi la directory che contiene il codice dell'applicazione di esempio alla `monorepo-sample` directory del repository GitHub AWS CodeCommit monorepo multi-pipeline triggers clonato.	Developer
Modificare il file `monorepo-main.json` .	Aggiungi il nome della directory del codice dell'applicazione e il nome della pipeline al file nel repository clonato. `monorepo-main.json`	Developer
Crea la pipeline.	Nella `Pipelines` directory del repository, aggiungi la pipeline `class` per la tua applicazione. La directory contiene due file di esempio. `pipeline_hotsite.py` `pipeline_demo.py` Ogni file ha tre fasi: origine, compilazione e distribuzione.	Developer

Operazione	Description	Competenze richieste
	È possibile copiare uno dei file e modificarlo in base ai requisiti dell'applicazione.	

Operazione	Description	Competenze richieste

Modificare il file `monorepo_config.py` .	In `service_map` , aggiungi il nome della directory per la tua applicazione e la classe che hai creato per la pipeline.
	
	Ad esempio, il codice seguente mostra una definizione di pipeline nella `Pipelines` directory che utilizza un file denominato `pipeline_mysample.py` con una `MySamplePipeline` classe:
	
	`...`
	`# Pipeline definition imports`
	`from pipelines.pipeline_demo import DemoPipeline`
	`from pipelines.pipeline_hotsite import HotsitePipeline`
	`from pipelines.pipeline_mysample import MySamplePipeline`
	
	`### Add your pipeline configuration here`
	`service_map: Dict[str, ServicePipeline] = {`
	`# folder-name -> pipeline-class`
	`'demo': DemoPipeline(),`
	`'hotsite': HotsitePipeline(),`
		Developer

Operazione	Description	Competenze richieste
	`'mysample': MySamplePipeline() }`	

Distribuisci lo stack MonoRepoStack

Operazione	Description	Competenze richieste
Distribuisci lo AWS CloudFormation stack.	Distribuisci lo AWS CloudFormation `MonoRepoStack` stack con i valori dei parametri predefiniti nella directory principale del repositor y clonato eseguendo il comando. `make deploy-core` È possibile modificare il nome del repository eseguendo il comando. `make deploy-core monorepo-name=<repo_name>` **Note** È possibile distribui re contempor aneamente entrambe le pipeline utilizzan do il comando. `make deploy monorepo-name=<rep o_name>`	Developer

Operazione	Description	Competenze richieste
Convalida il repository. CodeCommit	Verifica che le tue risorse siano state create eseguendo il comando. `aws codecommit get-repository --repository-name <repo_name>` **Important** Poiché lo CloudFormation stack crea il CodeCommit repository in cui è archiviato il monorepo, non eseguire il `cdk destroy MonoRepoStack` comando se hai iniziato a inserire modifiche al monorepo.	Developer
Convalida i risultati dello stack. CloudFormation	Verifica che lo CloudFormation `MonoRepoStack` stack sia stato creato e configurato correttamente eseguendo il comando seguente: `aws cloudformation list-stacks --stack-status-filter CREATE_COMPLETE --query 'StackSummaries[?StackName == 'MonoRepoStack']'`	Developer

Implementa lo stack PipelinesStack

Operazione	Description	Competenze richieste
Distribuisci lo CloudFormation stack.	Lo AWS CloudFormation `PipelinesStack` stack deve essere distribuito dopo aver distribuito lo stack. `MonoRepoStack` Lo stack aumenta di dimensioni quando vengono aggiunti nuovi microservizi alla base di codice di monorepo e viene ridistribuito quando viene integrato un nuovo microserv izio. Distribuisci lo stack Pipelines Stack eseguendo il comando. `make deploy-pipelines` **Note** È inoltre possibile distribuire contempor aneamente entrambe le pipeline eseguendo il comando. `make deploy monorepo-name=<rep o_name>` Il seguente output di esempio mostra come la `Pipelines Stacks` distribuzione stampa	Developer

Operazione	Description	Competenze richieste
	i dati URLs per i microservizi alla fine dell'implementazione: `Outputs: PipelinesStack.dem ourl = .cloudfront.net PipelinesStack.hotsi teurl = .cloudfro nt.net`	
Convalida i risultati dello AWS CloudFormation stack.	Verifica che lo AWS CloudFormation `Pipelines Stacks` stack sia stato creato e configurato correttam ente eseguendo il comando seguente: `aws cloudformation list-stacks --stack-s tatus-filter CREATE_CO MPLETE UPDATE_COMPLETE --query 'StackSum maries[?StackName == 'PipelinesStack']'`	Developer

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Elimina i tuoi AWS CloudForm ation stack.	Esegui il comando `make destroy`.	Developer
Elimina i bucket S3 per le tue pipeline.	1. Accedi Console di gestione AWSe apri la console Amazon Simple Storage Service (Amazon S3).	Developer

Operazione	Description	Competenze richieste
	2. Elimina i bucket S3 associati alle tue pipeline e usa il seguente nome: `pipelinesstack-codepipeline*`	

risoluzione dei problemi

Problema	Soluzione
Ho riscontrato dei problemi. AWS CDK	Consulta Risoluzione dei AWS CDK problemi più comuni nella documentazione di AWS CDK.
Ho inviato il mio codice di microservizio, ma la pipeline dei microservizi non ha funzionato.	Convalida della configurazione Verifica la configurazione della filiale: • Assicurati di inviare il codice al ramo corretto. Questa pipeline è configurata per funzionare solo quando vengono apportate modifiche al `main` ramo. I push verso altri rami non avviano la pipeline a meno che non siano configurati in modo specifico. • Dopo aver inviato il codice, controlla se il commit è visibile in AWS CodeCommit per assicurarti che il push abbia avuto successo e che la connessione tra l'ambiente locale e il repository sia intatta. Aggiorna le credenziali se ci sono problemi con il push del codice. Convalida i file di configurazione: • Verifica che la `service_map` variabile in rifletta `monorepo_config.py` accuratamente la struttura di directory corrente dei tuoi

Problema	Soluzione
	microservizi. Questa variabile svolge un ruolo cruciale nella mappatura del codice push sulla rispettiva pipeline. • Assicurati che `monorepo-main.json` sia aggiornato per includere la nuova mappatura per il tuo microservizio. Questo file è essenziale affinché la pipeline riconosca e gestisca correttamente le modifiche al microservizio. Risoluzione dei problemi sulla console AWS CodePipeline controlli: • In Console di gestione AWS, conferma di trovarti nel luogo in Regione AWS cui è ospitata la tua pipeline. Apri la CodePipeline console e controlla se la pipeline corrispon dente al tuo microservizio è stata avviata. Analisi degli errori: se la pipeline è stata avviata ma non è riuscita, esamina eventuali messaggi di errore o log forniti da CodePipel ine per capire cosa è andato storto. AWS Lambda risoluzione dei problemi: • Sulla AWS Lambda console, apri la funzione `monorepo-event-handler` Lambda. Verifica che la funzione sia stata avviata in risposta al codice push. Analisi dei log: esamina i log della funzione Lambda per eventuali problemi. I log possono fornire informazioni dettagliate su ciò che è accaduto durante l'esecuzione

Problema	Soluzione
	della funzione e aiutare a identificare se la funzione ha elaborato l'evento come previsto.

Problema	Soluzione
Devo ridistribuire tutti i miei microservizi.	Esistono due approcci per forzare la ridistrib uzione di tutti i microservizi. Scegliete l'opzione più adatta alle vostre esigenze. Approccio 1: Eliminare un parametro in Parameter Store Questo metodo prevede l'eliminazione di un parametro specifico in Systems Manager Parameter Store che tiene traccia dell'ulti mo ID di commit utilizzato per la distribuz ione. Quando si rimuove questo parametro , il sistema è costretto a ridistribuire tutti i microservizi al successivo trigger, perché lo percepisce come uno stato nuovo. Fasi: 1. Individuate la voce specifica del Parameter Store che contiene l'ID di commit o un indicatore di distribuzione correlato per il vostro monorepo. Il nome del parametro segue il formato: `"/MonoRepoTrigger/{repository}/{branch_name}/LastCommit"` 2. Valuta la possibilità di eseguire il backup del valore del parametro se è fondamentale o se desideri mantenere un record dello stato di distribuzione prima di reimpostarlo. 3. Utilizzate il Console di gestione AWS AWS CLI, o SDKs per eliminare il parametro identificato. Questa azione reimposta il marker di distribuzione. 4. Dopo l'eliminazione, il successivo invio al repository dovrebbe far sì che il sistema

Problema	Soluzione
	distribuisca tutti i microservizi, in quanto cerca il commit più recente da prendere in considerazione per la distribuzione. Pro: • Semplice e veloce da implementare con passaggi minimi. • Non richiede modifiche arbitrarie al codice per avviare le distribuzioni. Contro: • Controllo meno granulare sul processo di implementazione. • Potenzialmente rischioso se il Parameter Store viene utilizzato per gestire altre configurazioni critiche. Approccio 2: invia un commit in ogni sottocart ella monorepo Questo metodo prevede di apportare una modifica minore e di inserirla in ciascuna sottocartella di microservizi all'interno del monorepo per avviare le relative pipeline individuali. Fasi: 1. Elenca tutti i microservizi all'interno del monorepo che devono essere ridistribuiti. 2. Per ogni microservizio, apporta una modifica minima e senza impatto nella relativa sottocartella. Potrebbe trattarsi dell'aggi

Problema	Soluzione
	ornamento di un `README` file, dell'aggiunta di un commento in un file di configurazione o di qualsiasi modifica che non influisca sulla funzionalità del servizio. 3. Applica queste modifiche con un messaggio chiaro (ad esempio «Avvia la ridistribuzione dei microservizi») e inseriscile nell'archivio. Assicurati di inviare le modifiche al ramo che avvia la distribuzione. 4. Monitora le pipeline per ogni microserv izio per confermare che siano iniziate e completate correttamente. Pro: • Fornisce un controllo granulare su quali microservizi vengono ridistribuiti. • Più sicuro perché non comporta l'elimina zione di parametri di configurazione che potrebbero essere utilizzati per altri scopi. Contro: • Richiede più tempo, soprattutto con un gran numero di microservizi. • Richiede di apportare modifiche al codice non necessarie che potrebbero ingombrare la cronologia dei commit.

Risorse correlate

- Integrazione e distribuzione continue (CI/CD) con CDK Pipelines (documentazione)AWS CDK
- modulo aws-cdk/pipelines (riferimento API)AWS CDK

Integra un repository Bitbucket con AWS Amplify utilizzando AWS CloudFormation

Alwin Abraham, Amazon Web Services

Riepilogo

AWS Amplify ti aiuta a distribuire e testare rapidamente siti Web statici senza dover configurare l'infrastruttura normalmente richiesta. Puoi implementare l'approccio di questo modello se la tua organizzazione desidera utilizzare Bitbucket per il controllo del codice sorgente, sia per migrare il codice applicativo esistente che per creare una nuova applicazione. Utilizzando AWS CloudFormation per configurare automaticamente Amplify, offri visibilità sulle configurazioni che utilizzi.

Questo modello descrive come creare una pipeline e un ambiente di distribuzione front-end di integrazione continua e distribuzione continua (CI/CD) utilizzando AWS CloudFormation per integrare un repository Bitbucket con AWS Amplify. L'approccio del pattern significa che puoi creare una pipeline front-end Amplify per implementazioni ripetibili.

Prerequisiti e limitazioni

Prerequisiti

- Un account Amazon Web Services (AWS) attivo
- Un account Bitbucket attivo con accesso da amministratore
- Accesso a un terminale che utilizza cURL o l'applicazione Postman
- Familiarità con Amplify
- Familiarità con AWS CloudFormation
- Familiarità con i file in formato YAML

Architecture

Stack tecnologico

- Amplify

- AWS CloudFormation
- Bitbucket

Tools (Strumenti)

- AWS Amplify — Amplify aiuta gli sviluppatori a sviluppare e distribuire app mobili e Web basate sul cloud.
- AWS CloudFormation: AWS CloudFormation è un servizio che ti aiuta a modellare e configurare le tue risorse AWS in modo da poter dedicare meno tempo alla gestione di tali risorse e più tempo a concentrarti sulle applicazioni eseguite in AWS.
- Bitbucket — Bitbucket è una soluzione di gestione di repository Git progettata per team di professionisti. Ti offre un posto centrale per gestire gli archivi Git, collaborare sul codice sorgente e guidarti attraverso il flusso di sviluppo.

Codice

Il `bitbucket-amplify.yml` file (allegato) contiene il CloudFormation modello AWS per questo modello.

Epiche

Configura il repository Bitbucket

Operazione	Description	Competenze richieste
(Facoltativo) Crea un repository Bitbucket.	1. Accedi al tuo account Bitbucket e crea un nuovo repository. Per ulteriori informazioni su questo argomento, consulta Creare un repository Git nella documentazione di Bitbucket. 2. Registra il nome dell'area di lavoro.	DevOps ingegnere

Operazione	Description	Competenze richieste
	Note Puoi anche utilizzare un repository Bitbucket esistente.	
Apri le impostazioni dell'area di lavoro.	1. Apri l'area di lavoro e scegli la scheda Repository. 2. Scegli il repository che desideri integrare con Amplify. 3. Scegli il nome dell'area di lavoro che si trova sopra il nome del repository. 4. Nella barra laterale, scegli Impostazioni.	DevOps ingegnere

Operazione	Description	Competenze richieste
Crea un OAuth consumatore.	1. Nella sezione App e funzionalità, scegli OAuth consumatori, quindi scegli Aggiungi consumatore. 2. Inserisci un nome per il tuo consumatore, ad esempio`Amplify Integration` . 3. Inserisci un URL di callback. Sebbene questo campo sia un input obbligatorio, non viene utilizzato per completar e l'integrazione, quindi il valore potrebbe essere `http://localhost:3000` 4. Seleziona la casella Questo è un consumatore privato. 5. Scegli le seguenti autorizza zioni: • Progetto — `Read` • Archivi — `Admin` • Richieste pull — `Read` • Webhook — e `Read Write` 6. Lascia le scelte predefini te per tutti gli altri campi e scegli Invia. 7. Registra la chiave e il segreto generati.	DevOps ingegnere

Operazione	Description	Competenze richieste
Ottieni il token di OAuth accesso.	1. Apri una finestra del terminale ed esegui il comando seguente: `curl -X POST -u "KEY:SECRET" https://bitbucket.org/site/oauth2/access_token -d grant_type=client_credentials` **Important** Sostituisci `KEY` e `SECRET` con la chiave e il segreto che hai registrato in precedenza. 2. Registra il token di accesso senza usare le virgolette. Il token è valido solo per un periodo di tempo limitato e il tempo predefinito è di due ore. È necessario eseguire il CloudFormation modello AWS in questo lasso di tempo.	DevOps ingegnere

Crea e distribuisci lo stack AWS CloudFormation

Operazione	Description	Competenze richieste
Scarica il CloudFormation modello AWS.	Scarica il CloudFormation modello `bitbucket-amplify.yml` AWS (allegato). Questo modello crea la CI/CD pipeline in Amplify, oltre al progetto e al ramo Amplify.	
Crea e distribuisci lo CloudFormation stack AWS.	1. Accedi alla Console di gestione AWS nella regione AWS in cui desideri effettuare la distribuzione e apri la CloudFormation console AWS. 2. Scegli Create Stack (con nuove risorse), quindi scegli Carica un file modello. 3. Caricare il file `bitbucket-amplify.yml` 4. Scegli Avanti, inserisci il nome dello stack, quindi inserisci i seguenti parametri: • Token di accesso: incolla il token di OAuth accesso che hai creato in precedenza. • URL del repository: aggiungi l'URL del repository del progetto Bitbucket. L'URL è in genere nel seguente	DevOps ingegnere

Operazione	Description	Competenze richieste
	formato: `https://bitbucket.org/<WORKSPACE_NAME>/<REPO_NAME>` • Nome del ramo: deve corrispondere al nome di un ramo nel tuo repository Bitbucket. Non è necessario che questo ramo esista quando esegui lo CloudFormation stack AWS, ma è necessario per distribuire il codice nell'ambiente. • Nome del progetto: questo è il nome da associare al progetto Amplify. 5. Scegli Avanti, quindi scegli Crea pila.	

Prova la CI/CD pipeline

Operazione	Description	Competenze richieste
Distribuisci il codice nella filiale del tuo repository.	1. Clona il tuo repository Bitbucket eseguendo il seguente comando: `git clone https://bitbucket.org/<WORKSPACE_NAME>/<REPO_NAME>`	Sviluppatore di app

Operazione	Description	Competenze richieste
	2. Scopri il nome del ramo utilizzato durante l'esecuzione dello CloudFormation script AWS. Per creare e controllare un nuovo ramo, esegui il `git checkout -b <BRANCH_NAME>` comando. Per estrarre un ramo esistente, esegui il `git checkout <BRANCH_NAME>` comando 3. Inserisci il codice nel ramo e invialo al ramo remoto eseguendo `git push` i comandi `git commit` and. 4. Amplify quindi crea e distribuisce l'applicazione. Per ulteriori informazioni su questo argomento, consulta i comandi Git di base nella documentazione di Bitbucket.	

Risorse correlate

Metodi di autenticazione (documentazione Atlassian)

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Avvia un CodeBuild progetto su più account AWS utilizzando Step Functions e una funzione proxy Lambda

Richard Milner-Watts e Amit Anjarlekar, Amazon Web Services

Riepilogo

Questo modello dimostra come avviare in modo asincrono un progetto AWS CodeBuild su più account AWS utilizzando AWS Step Functions e una funzione proxy AWS Lambda. Puoi utilizzare la macchina a stati Step Functions di esempio del pattern per testare il successo del tuo CodeBuild progetto.

CodeBuild ti aiuta ad avviare attività operative utilizzando l'AWS Command Line Interface (AWS Command Line Interface) da un ambiente di runtime completamente gestito. Puoi modificare il comportamento del CodeBuild progetto in fase di esecuzione sovrascrivendo le variabili di ambiente. Inoltre, puoi utilizzarlo CodeBuild per gestire i flussi di lavoro. Per ulteriori informazioni, consulta Service Catalog Tools sul sito Web di AWS Workshop e Schedule jobs in Amazon RDS for PostgreSQL CodeBuild using AWS EventBridge and Amazon sul blog di AWS Database.

Prerequisiti e limitazioni

Prerequisiti

- Due account AWS attivi: un account di origine per richiamare una funzione proxy Lambda con Step Functions e un account di destinazione per la creazione di un progetto di esempio remoto CodeBuild

Limitazioni

- Questo modello non può essere utilizzato per copiare artefatti tra account.

Architecture

Il diagramma seguente mostra l'architettura creata da questo modello.

Il diagramma mostra il flusso di lavoro seguente:

1. La macchina a stati Step Functions analizza la mappa di input fornita e richiama la funzione proxy Lambda (`codebuild-proxy-lambda`) per ogni account, regione e progetto definito.
2. La funzione proxy Lambda utilizza AWS Security Token Service (AWS STS) per assumere un ruolo proxy IAM (`codebuild-proxy-role`), associato a una policy IAM (`codebuild-proxy-policy`) nell'account di destinazione.
3. Utilizzando il ruolo assunto, la funzione Lambda avvia il CodeBuild progetto e restituisce l'ID del CodeBuild lavoro. La macchina a stati Step Functions esegue un loop ed esegue il polling del CodeBuild lavoro fino a quando non riceve uno stato di successo o di fallimento.

La logica della macchina a stati è mostrata nell'immagine seguente.

Stack tecnologico

- AWS CloudFormation
- CodeBuild
- IAM
- Lambda
- Step Functions
- X-Ray

Tools (Strumenti)

- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account e regioni AWS.
- AWS CloudFormation Designer fornisce un editor JSON e YAML integrato che ti aiuta a visualizzare e modificare i modelli. CloudFormation
- AWS CodeBuild è un servizio di build completamente gestito che ti aiuta a compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.

- AWS Step Functions è un servizio di orchestrazione serverless che ti aiuta a combinare le funzioni di AWS Lambda e altri servizi AWS per creare applicazioni aziendali critiche.
- AWS X-Ray ti aiuta a raccogliere dati sulle richieste servite dalla tua applicazione e fornisce strumenti che puoi utilizzare per visualizzare, filtrare e ottenere informazioni su tali dati per identificare problemi e opportunità di ottimizzazione.

Codice

Il codice di esempio per questo modello è disponibile nel repository GitHub Cross Account CodeBuild Proxy. Questo modello utilizza la libreria AWS Lambda Powertools for Python per fornire funzionalità di registrazione e tracciamento. Per ulteriori informazioni su questa libreria e le sue utilità, consulta Powertools for AWS Lambda (Python).

Best practice

1. Modifica i valori del tempo di attesa nella macchina a stati Step Function per ridurre al minimo le richieste di polling sullo stato del lavoro. Utilizza il tempo di esecuzione previsto per il CodeBuild progetto.
2. Modifica la `MaxConcurrency` proprietà della mappa in Step Functions per controllare quanti CodeBuild progetti possono essere eseguiti in parallelo.
3. Se necessario, esaminate il codice di esempio per verificarne la disponibilità alla produzione. Valuta quali dati potrebbero essere registrati dalla soluzione e se la CloudWatch crittografia Amazon predefinita è sufficiente.

Epiche

Crea la funzione proxy Lambda e il ruolo IAM associato nell'account di origine

Operazione	Description	Competenze richieste
Registra l'account AWS IDs.	IDs Gli account AWS sono necessari per configurare l'accesso tra account. Registra l'ID dell'account AWS per i tuoi account di origine e di destinazione. Per ulteriori	AWS DevOps

Operazione	Description	Competenze richieste
	informazioni, consulta Finding your AWS account ID nella documentazione IAM.	
Scarica i CloudFormation modelli AWS.	1. Scarica il CloudFormation modello `sample_target_codebuild_template.yaml` AWS dal GitHub repository per questo modello. 2. Scarica il CloudFormation modello `codebuild_lambda_proxy_template.yaml` AWS dal GitHub repository per questo modello. **Note** Nei CloudFormation modelli AWS, `<SourceAccountId>` è l'ID dell'account AWS per l'account di origine e `<TargetAccountId>` l'ID dell'account AWS per l'account di destinazione.	AWS DevOps

Operazione	Description	Competenze richieste
Crea e distribuisci lo CloudFormation stack AWS.	1. Accedi alla Console di gestione AWS per il tuo account di origine, apri la CloudFormation console AWS e scegli Stacks. 2. Scegliere Create stack (Crea stack), quindi With new resources (standard) (Con nuove risorse (standard)). 3. Come Template source (Origine modello), scegliere Upload a template file (Carica un file di modello). 4. Per Carica un file modello, scegli file, quindi scegli il `codebuild_lambda_proxy_template.yaml` file scaricato. Scegli Next (Successivo). 5. Per Nome dello stack, inserisci un nome per lo stack (ad esempio,`codebuild-lambda-proxy`). 6. **Note** Sostituite il `crossAccountTargetRoleArn` parametro con il vostro <TargetAc	AWS DevOps

Operazione	Description	Competenze richieste
	countId> (ad esempio,<arn:aws:iam::123456789012:role/proxy-lambda-codebuild-role>). : Non è necessario aggiornare il valore predefinito per il **targetCodeBuildProject** parametro. 7. Scegliete Avanti, accettate le opzioni predefinite per la creazione dello stack, quindi scegliete Avanti. 8. Seleziona la casella di controllo Riconosco che AWS CloudFormation potrebbe creare risorse IAM con nomi personalizzati, quindi scegli Create stack. **Note** È necessario creare lo CloudFormation stack AWS per la funzione proxy Lambda prima di creare risorse negli account di destinazi	

Operazione	Description	Competenze richieste
	one. Quando crei una policy di fiducia in un account di destinazi one, il ruolo IAM viene tradotto dal nome del ruolo a un identific atore interno. Questo è il motivo per cui il ruolo IAM deve già esistere.	
Conferma la creazione della funzione proxy e della macchina a stati.	1. Attendi che lo CloudForm ation stack AWS raggiunga lo stato CREATE_CO MPLETE. Questa operazion e dovrebbe richiedere meno di un minuto. 2. Apri la console AWS Lambda, scegli Funzioni, quindi trova la `lambda-pr oxy-ProxyLambda-<G UID>` funzione. 3. Apri la console AWS Step Functions, scegli le macchine a stati, quindi trova la macchina a `sample-crossaccoun t-codebuild-state- machine` stati.	AWS DevOps

Crea un ruolo IAM nell'account di destinazione e avvia un CodeBuild progetto di esempio

Operazione	Description	Competenze richieste
Crea e distribuisci lo CloudFormation stack AWS.	1. Accedi alla Console di gestione AWS per il tuo account di destinazione, apri la CloudFormation console AWS e scegli Stacks. 2. Scegli Create Stack, quindi scegli Con nuove risorse (standard). 3. Come Template source (Origine modello), scegliere Upload a template file (Carica un file di modello). 4. Per Carica un file modello, scegli Scegli file, quindi scegli il `sample_target_codebuild_template.yaml` file. Scegli Next (Successivo). 5. Per Nome dello stack, inserisci un nome per lo stack (ad esempio:`sample-codebuild-stack`). 6. Sostituite il `crossAccountSourceRoleArn` parametro con il vostro `<SourceAccountId>` (ad esempio,`<arn:aws:iam::123456789012:role/codebuild-proxy-lambda-role>`).	AWS DevOps

Operazione	Description	Competenze richieste
	7. Scegliete Avanti, accettate le opzioni predefinite per la creazione dello stack, quindi scegliete Avanti. 8. Seleziona la casella di controllo Riconosco che AWS CloudFormation potrebbe creare risorse IAM con nomi personalizzati, quindi scegli Create stack.	
Verifica la creazione del CodeBuild progetto di esempio.	1. Attendi che lo CloudFormation stack AWS raggiunga lo stato CREATE_COMPLETE. Questa operazione dovrebbe richiedere meno di un minuto. 2. Apri la CodeBuild console AWS e trova il `sample-codebuild-project` progetto.	AWS DevOps

Prova la funzione proxy Lambda per più account

Operazione	Description	Competenze richieste
Avvia la macchina statale.	1. Accedi alla Console di gestione AWS per il tuo account di origine, apri la console AWS Step Functions e scegli Macchine a stati. 2. Scegli la macchina a `sample-crossaccoun`	AWS DevOps

Operazione	Description	Competenze richieste
	`t-codebuild-state-machine` stati, quindi scegli Avvia esecuzione. 3. Nell'editor di Input, inserisci il seguente codice JSON e `<TargetAccountID>` sostituiscilo con l'ID dell'account AWS che contiene il CodeBuild progetto. `{ "crossAccountTargetRoleArns": [ { "arn": "arn:aws:iam::<TargetAccountID>:role/proxy-lambda-codebuild-role", "region": "eu-west-1", "codeBuildProject": "sample-codebuild-project", "SampleValue1": "Value1", "SampleValue2": "Value2" } ] }` **Note** Le coppie chiave-valore vengono passate come	

Operazione	Description	Competenze richieste
	variabili di ambiente dalla funzione nell'account di origine al CodeBuild progetto nell'acco unt di destinazione. 4. Selezionare Start execution (Avvia esecuzione). 5. Nella scheda Dettagli della pagina della macchina a stati, controlla se lo stato di esecuzione è impostato su Riuscito. Ciò conferma che la macchina a stati è in esecuzione. Nota: possono essere necessari circa 30 secondi prima che la macchina a stati raggiunga lo stato Riuscito. 6. Per visualizzare l'output e l'input di un passaggio nella macchina a stati, espandi quel passaggio nella sezione Cronologia degli eventi di esecuzion e. Ad esempio, espandi il passaggio Lambda - CodeBuild Proxy — Start. L'output include dettagli sulle variabili di ambiente sostituite, sul payload originale e sull'ID del lavoro. CodeBuild	

Operazione	Description	Competenze richieste
Convalida le variabili di ambiente.	1. Accedi alla Console di gestione AWS per il tuo account di destinazione. 2. Apri la CodeBuild console AWS, espandi Build, quindi scegli Build projects. 3. Scegli il `sample-codebuild-project` progetto, quindi scegli Visualizza dettagli. 4. Nella scheda Cronologia build, scegli la build più recente del progetto, quindi scegli Visualizza registri. 5. Nell'output del registro, verifica che le variabili di ambiente stampate su STDOUT corrispondano alle variabili di ambiente della macchina a stati di esempio Step Functions.	AWS DevOps

risoluzione dei problemi

Problema	Soluzione
L'esecuzione di Step Functions sta impiegando più tempo del previsto.	Regola la `MaxConcurrency` proprietà della mappa nella macchina a stati Step Function per controllare quanti CodeBuild progetti possono essere eseguiti in parallelo.
L'esecuzione dei CodeBuild lavori richiede più tempo del previsto.	1. Regola i valori del tempo di attesa nella macchina a stati Step Functions per ridurre

Problema	Soluzione
	al minimo le richieste di polling sullo stato del lavoro. Utilizza il tempo di esecuzione previsto per il CodeBuild progetto. 2. Valuta se CodeBuild è lo strumento appropriato da utilizzare. Ad esempio, il tempo necessario per inizializzare un CodeBuild job può essere significativamente più lungo di AWS Lambda. Se sono richiesti un throughput elevato e tempi di completam ento rapidi, prendi in considerazione la migrazione della logica di business su AWS Lambda e l'utilizzo di un'architettura fan-out.

Gestione del failover Multi-AZ per i cluster EMR utilizzando Application Recovery Controller

Aarti Rajput, Ashish Bhatt, Neeti Mishra e Nidhi Sharma, Amazon Web Services

Riepilogo

Questo modello offre una strategia di disaster recovery efficiente per i carichi di lavoro di Amazon EMR per contribuire a garantire un'elevata disponibilità e coerenza dei dati su più zone di disponibilità all'interno di un'unica. Regione AWS Il design utilizza Amazon Application Recovery Controller e un Application Load Balancer per gestire le operazioni di failover e la distribuzione del traffico per un cluster EMR basato su Apache Spark.

In condizioni standard, la zona di disponibilità principale ospita un cluster EMR attivo e un'applicazione con funzionalità complete read/write . Se una zona di disponibilità si guasta inaspettatamente, il traffico viene reindirizzato automaticamente alla zona di disponibilità secondaria, dove viene avviato un nuovo cluster EMR. Entrambe le zone di disponibilità accedono a un bucket Amazon Simple Storage Service (Amazon S3) condiviso tramite endpoint gateway dedicati, che garantiscono una gestione coerente dei dati. Questo approccio riduce al minimo i tempi di inattività e consente il ripristino rapido dei carichi di lavoro critici relativi ai Big Data durante i guasti delle zone di disponibilità. La soluzione è utile in settori come la finanza o la vendita al dettaglio, dove l'analisi in tempo reale è fondamentale.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Amazon EMR su Amazon Elastic Compute Cloud (Amazon) EC2
- Accesso dal nodo master del cluster EMR ad Amazon S3.
- AWS Infrastruttura Multi-AZ

Limitazioni

- Alcune Servizi AWS non sono disponibili in tutte Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- Amazon EMR 6.x e versioni successive

Architecture

Stack tecnologico Target

- Cluster Amazon EMR
- Amazon Application Recovery Controller
- Application Load Balancer
- Bucket Amazon S3
- Endpoint gateway per Amazon S3

Architettura di destinazione

Questa architettura fornisce la resilienza delle applicazioni utilizzando più zone di disponibilità e implementando un meccanismo di ripristino automatico tramite Application Recovery Controller.

1. L'Application Load Balancer indirizza il traffico verso l'ambiente Amazon EMR attivo, che in genere è il cluster EMR primario nella zona di disponibilità principale.
2. Il cluster EMR attivo elabora le richieste delle applicazioni e si connette ad Amazon S3 tramite l'endpoint gateway Amazon S3 dedicato per le operazioni di lettura e scrittura.
3. Amazon S3 funge da archivio centrale di dati ed è potenzialmente utilizzato come checkpoint o come storage condiviso tra cluster EMR. I cluster EMR mantengono la coerenza dei dati quando scrivono direttamente su Amazon S3 tramite il protocollo `s3://` e l'EMR File System (EMRFS).
4. Application Recovery Controller monitora continuamente lo stato della zona di disponibilità principale e gestisce automaticamente le operazioni di failover quando necessario.
5. Se l'Application Recovery Controller rileva un errore nel cluster EMR primario, esegue le seguenti azioni:
   - Avvia il processo di failover sul cluster EMR secondario nella zona di disponibilità 2.
   - Aggiorna le configurazioni di routing per indirizzare il traffico verso il cluster secondario.

Tools (Strumenti)

Servizi AWS

- Amazon Application Recovery Controller ti aiuta a gestire e coordinare il ripristino delle tue applicazioni su tutte Regioni AWS le zone di disponibilità. Questo servizio semplifica il processo e migliora l'affidabilità del ripristino delle applicazioni riducendo i passaggi manuali richiesti dagli strumenti e dai processi tradizionali.
- Application Load Balancer opera a livello di applicazione, che è il settimo livello del modello Open Systems Interconnection (OSI). Distribuisce il traffico delle applicazioni in entrata su più destinazioni, ad esempio EC2 istanze, in più zone di disponibilità. Ciò aumenta la disponibilità dell'applicazione.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando.
- Amazon EMR è una piattaforma di big data che fornisce elaborazione dati, analisi interattiva e apprendimento automatico per framework open source come Apache Spark, Apache Hive e Presto.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon S3 offre una semplice interfaccia di servizio Web che puoi utilizzare per archiviare e recuperare qualsiasi quantità di dati, in qualsiasi momento e da qualsiasi luogo. Utilizzando questo servizio, puoi creare facilmente applicazioni che utilizzano lo storage nativo del cloud.
- Gli endpoint gateway per Amazon S3 sono gateway specificati nella tabella di routing per accedere ad Amazon S3 dal tuo cloud privato virtuale (VPC) sulla rete. AWS

Best practice

- Segui le AWS best practice per la sicurezza, l'identità e la conformità per garantire un'architettura solida e sicura.
- Allinea l'architettura con il AWS Well-Architected Framework.
- Usa Amazon S3 Access Grants per gestire l'accesso dal tuo cluster EMR basato su Spark ad Amazon S3. Per maggiori dettagli, consulta il post del blog Usa Amazon EMR con S3 Access Grants per scalare l'accesso Spark ad Amazon S3.
- Migliora le prestazioni di Spark con Amazon S3.

Epiche

Configurare l'ambiente

Operazione	Description	Competenze richieste
Accedi alla Console di gestione AWS.	Accedere alla Console di gestione AWS come utente IAM. Per istruzioni, consulta la AWS documentazione.	AWS DevOps
Configura il AWS CLI.	Installa AWS CLI o aggiornalo alla versione più recente in modo da poter interagire con Servizi AWS in Console di gestione AWS. Per istruzioni, consulta la AWS CLI documentazione.	AWS DevOps

Implementa un'applicazione Spark sul tuo cluster EMR

Operazione	Description	Competenze richieste
Crea un bucket S3.	1. Crea un bucket S3 per archiviare il set di dati di input, i log, l'applicazione e i dati di output. Per istruzioni, consulta la documentazione di Amazon S3. 2. Organizza il bucket in cartelle separate per i dati di input (`dataset`), i log (`logs`), l'applicazione Spark () e i dati di output (`spark-app `). `output`	AWS DevOps

Operazione	Description	Competenze richieste

Crea un cluster EMR.	1. Utilizza i seguenti AWS CLI comandi per creare un cluster EMR (ad esempio, versione 6.12 o successiva) con istanze che si estendono su due zone di disponibilità (ad esempio `us-east-1a us-east-1b`) per un'elevata disponibilità. Il comando specifica il tipo di istanza come `m4.large` esempio.
	
	`aws emr create-cluster \ --ec2-attributes AvailabilityZone=<AZ-name-1> \ --release-label emr-6.12.0 \ --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large`
	
	`aws emr create-cluster \ --ec2-attributes AvailabilityZone=<AZ-name-2> \ --release-label emr-6.12.0 \ --instance-groups InstanceGroupType=`
		AWS DevOps

Operazione	Description	Competenze richieste

	<pre>MASTER,InstanceCou
	nt=1,InstanceType=
	m4.large InstanceG
	roupType=CORE,Inst
	anceCount=2,Instan
	ceType=m4.large</pre>
	Per ulteriori informazioni, consulta il comando create-cluster e la documentazione di Amazon EMR.
	2. Fornisci alla coppia di chiavi, al ruolo di servizio e al profilo dell'istanza le autorizzazioni richieste, se necessario.	

Operazione	Description	Competenze richieste
Configurare le impostazioni di sicurezza per il cluster EMR.	1. Identifica il gruppo di sicurezza associato al nodo principale del cluster EMR utilizzando il comando AWS CLI describe-cluster: `aws emr describe-cluster --cluster-id j-XXXXXXXX` 2. Per migliorare la sicurezza, modifica le impostazioni del gruppo di sicurezza per consentire l'accesso SSH (porta TCP 22) al nodo master, ma limitalo al tuo indirizzo IP specifico. Per ulteriori informazioni, consulta la documentazione di Amazon EMR.	AWS DevOps

Operazione	Description	Competenze richieste

Connect al cluster EMR.	Connect al nodo master del cluster EMR tramite SSH utilizzando la key pair fornita.
	
	Assicurati che il file key pair sia presente nella stessa directory dell'applicazione.
	
	Esegui i seguenti comandi per impostare le autorizza zioni corrette per la key pair e stabilire la connessione SSH:
	
	`chmod 400 <key-pair-name>`
	`ssh -i ./<key-pair-name> hadoop@<master-node-public-dns>`
		AWS DevOps

Operazione	Description	Competenze richieste

Distribuisci l'applicazione Spark.	Dopo aver stabilito la connessione SSH, ti troverai nella console Hadoop.
	
	1. Crea o modifica il file dell'applicazione Spark (`main.py`) utilizzando un editor di testo come vim:
	
	`vim main.py`
	
	Per ulteriori informazioni sulla creazione e la modifica dell'applicazione Spark, consulta la documentazione di Amazon EMR.
	2. Invia l'applicazione Spark al cluster EMR, specificando i dati di input e le posizioni dei dati di output nel bucket S3:
	
	`spark-submit main.py –data_source <input-data-folder-in-s3> –output_uri <output-folder-in-s3>`
	
	Ecco un esempio basato sulle cartelle che hai configurato in precedenza:
		AWS DevOps

Operazione	Description	Competenze richieste

	`spark-submit main.py –data_source dataset –output_uri output`
	3. Monitora lo stato di avanzamento dell'applicazione controllando i registri dell'applicazione:
	`yarn logs -applicationId <application-id>`	

Monitora l'applicazione Spark.	1. Apri un'altra finestra di terminale e stabilisci un tunnel SSH verso l'interfaccia utente web del gestore delle risorse del cluster EMR: `ssh -i <key-pair-name> -N -L 8157:<resource-manager-public-dns>:8088 hadoop@<resource-manager-public-dns>` 2. Per monitorare l'applicazione, accedi all'interfaccia utente web del gestore delle risorse accedendo al `http://localhost:8157` tuo browser web.	AWS DevOps

Sposta il traffico verso un'altra zona di disponibilità

Operazione	Description	Competenze richieste
Crea un Application Load Balancer.	Configura il gruppo target che indirizza il traffico tra i nodi master di Amazon EMR distribuiti su due zone di disponibilità all'interno di una. Regione AWS Per istruzioni, consulta Creare un gruppo target per il tuo Application Load Balancer nella documentazione di Elastic Load Balancing.	AWS DevOps
Configura lo spostamento zonale in Application Recovery Controller.	In questo passaggio, utilizzerai la funzionalità di spostamento zonale in Application Recovery Controller per spostare il traffico verso un'altra zona di disponibilità. 1. Apri la console Application Recovery Controller. 2. In Guida introduttiva, scegli Spostamento zonale, Avvia spostamento zonale. 3. Seleziona la zona di disponibilità da cui desideri allontanare il traffico. 4. Seleziona una risorsa supportata (ad esempio Application Load Balancer) per lo spostamento zonale dalla tabella Risorse.	AWS DevOps

Operazione	Description	Competenze richieste
	5. Per Imposta la scadenza del turno zonale, scegli o inserisci una scadenza per il turno zonale. È possibile impostare una durata compresa tra 1 minuto e tre giorni (72 ore). Tutti gli spostamenti zonali sono temporanei. È necessario impostare una scadenza, ma è possibile aggiornare i turni attivi in un secondo momento per impostare un nuovo periodo di scadenza fino a tre giorni. 6. Inserisci un commento su questo spostamento zonale. 7. Seleziona la casella di controllo per confermar e che l'avvio di uno spostamento zonale ridurrà la capacità disponibile per l'applicazione spostando il traffico lontano dalla zona di disponibilità. 8. Scegli Avvia. Per utilizzare il AWS CLI, consulta Esempi di utilizzo di AWS CLI with zonal shift nella documentazione di Application Recovery Controller.	

Operazione	Description	Competenze richieste

Verifica la configurazione e l'avanzamento dei turni zonali.	1. Verifica le risorse registrate con Zonal Shift:
	`aws arc-zonal-shift list-managed-resources --region <AWS-region-name>`
	Ad esempio, l'output seguente conferma che le risorse sono attive e funzionanti in entrambe le zone di disponibilità.
	`"appliedWeights": { "use1-az1": 1.0, "use1-az2": 1.0 },`
	2. Per visualizzare lo spostamento zonale, utilizzate il seguente AWS CLI comando per avviare lo spostamento zonale:
	`aws arc-zonal-shift start-zonal-shift \ --resource-identifier <application-load-balancer-arn> \ --away-from <source-AZ> \ --expires-in 10m --comment "testing" \ --region <AWS-region-name>`
		AWS DevOps

Operazione	Description	Competenze richieste

	dove `<source-AZ>` è l'identificatore della zona di disponibilità da cui vuoi allontanare il traffico ed `<application-load-balancer-arn>` è l'Amazon Resource Name (ARN) del tuo Application Load Balancer.
	3. Verifica che il traffico si sia spostato verso un'altra zona di disponibilità.
	`aws arc-zonal-shift get-managed-resource \ --resource-identifier <application-load-balancer-arn> \ --region <AWS-region-name>`
	Puoi vedere lo spostamento zonale confermato da questi pesi:
	`"appliedWeights": { "use1-az1": 0.0, "use1-az2": 1.0 },`	

Risorse correlate

- AWS CLI comandi:
  - create-cluster
  - describe-cluster
  - arc-zonal-shift
- Configurazione dei tipi di istanze del cluster Amazon EMR e delle best practice per le istanze Spot (documentazione Amazon EMR)
- Le migliori pratiche di sicurezza in IAM (documentazione IAM)
- Usa i profili di istanza (documentazione IAM)
- Usa zonal shift e zonal autoshift per ripristinare le applicazioni in ARC (documentazione dell'Application Recovery Controller)

Gestisci le blue/green distribuzioni di microservizi su più account e regioni utilizzando i servizi di codice AWS e le chiavi multiregionali AWS KMS

Balaji Vedagiri, Vanitha Dontireddy, Ashish Kumar, Faisal Shahdad, Vivek Thangamuthu e Anand Krishna Varanasi, Amazon Web Services

Riepilogo

Questo modello descrive come distribuire un'applicazione globale di microservizi da un account AWS centrale a più account di carico di lavoro e regioni in base a una blue/green strategia di distribuzione. Il pattern supporta quanto segue:

- Il software è sviluppato in un account centrale, mentre i carichi di lavoro e le applicazioni sono distribuiti su più account e regioni AWS.
- Una singola chiave multiregionale AWS Key Management System (AWS KMS) viene utilizzata per la crittografia e la decrittografia per coprire il disaster recovery.
- La chiave KMS è specifica della regione e deve essere gestita o creata in tre diverse regioni per gli artefatti della pipeline. Una chiave multiregionale KMS aiuta a mantenere lo stesso ID chiave in tutte le regioni.
- Il modello di ramificazione del flusso di lavoro Git è implementato con due rami (development e main) e il codice viene unito utilizzando pull request (PRs). La funzione AWS Lambda distribuita da questo stack crea un PR dal ramo di sviluppo al ramo principale. L'unione delle pubbliche relazioni con la filiale principale avvia una CodePipeline pipeline AWS, che orchestra il flusso di integrazione continua e distribuzione continua (CI/CD) e distribuisce gli stack tra gli account.

Questo modello fornisce un esempio di configurazione dell'infrastruttura come codice (IaC) tramite gli CloudFormation stack AWS per dimostrare questo caso d'uso. La blue/green distribuzione dei microservizi viene implementata utilizzando AWS CodeDeploy.

Prerequisiti e limitazioni

Prerequisiti

- Quattro account AWS attivi:
  - Un account di strumenti per gestire la pipeline di codice e mantenere il CodeCommit repository AWS.
  - Tre account di carico di lavoro (test) per la distribuzione del carico di lavoro dei microservizi.

- Questo modello utilizza le seguenti regioni. Se desideri utilizzare altre regioni, devi apportare le modifiche appropriate agli stack multiregione AWS CodeDeploy e AWS KMS.
  - Account Tools (AWS CodeCommit): `ap-south-1`
  - Account Workload (test) 1: `ap-south-1`
  - Account per il carico di lavoro (test) 2: `eu-central-1`
  - Account per il carico di lavoro (test) 3: `us-east-1`
- Tre bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) per le regioni di distribuzione in ogni account di carico di lavoro. (Queste vengono chiamate `S3BUCKETNAMETESTACCOUNT1` `S3BUCKETNAMETESTACCOUNT2` e `S3BUCKETNAMETESTACCOUNT3` più avanti in questo schema).

  Ad esempio, puoi creare questi bucket in account e regioni specifici con nomi di bucket univoci come segue (sostituisci xxxx con un numero casuale):

```
##In Test Account 1
aws s3 mb s3://ecs-codepipeline-xxxx-ap-south-1 --region ap-south-1
##In Test Account 2
aws s3 mb s3://ecs-codepipeline-xxxx-eu-central-1 --region eu-central-1
##In Test Account 3
aws s3 mb s3://ecs-codepipeline-xxxx-us-east-1 --region us-east-1

#Example
##In Test Account 1
aws s3 mb s3://ecs-codepipeline-18903-ap-south-1 --region ap-south-1
##In Test Account 2
aws s3 mb s3://ecs-codepipeline-18903-eu-central-1 --region eu-central-1
##In Test Account 3
aws s3 mb s3://ecs-codepipeline-18903-us-east-1 --region us-east-1
```

Limitazioni

Il modello utilizza AWS CodeBuild e altri file di configurazione per distribuire un microservizio di esempio. Se hai un tipo di carico di lavoro diverso (ad esempio, serverless), devi aggiornare tutte le configurazioni pertinenti.

Architecture

Stack tecnologico Target

- AWS CloudFormation
- AWS CodeCommit
- AWS CodeBuild
- AWS CodeDeploy
- AWS CodePipeline

Architettura Target

Automazione e scalabilità

La configurazione è automatizzata utilizzando i modelli di CloudFormation stack AWS (IaC). Può essere facilmente scalato per più ambienti e account.

Tools (Strumenti)

Servizi AWS

- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account e regioni AWS.
- AWS CodeBuild è un servizio di build completamente gestito che ti aiuta a compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS CodeCommit è un servizio di controllo delle versioni che consente di archiviare e gestire archivi Git in modo privato, senza dover gestire il proprio sistema di controllo del codice sorgente.
- AWS CodeDeploy automatizza le distribuzioni su Amazon Elastic Compute Cloud (Amazon EC2) o istanze locali, funzioni AWS Lambda o servizi Amazon Elastic Container Service (Amazon ECS).
- AWS ti CodePipeline aiuta a modellare e configurare rapidamente le diverse fasi di un rilascio di software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- Amazon Elastic Container Service (Amazon ECS) è un servizio rapido e scalabile di gestione dei container che ti aiuta a eseguire, arrestare e gestire container in un cluster.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.

- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Strumenti aggiuntivi

- Git è un sistema di controllo delle versioni distribuito e open source che funziona con il CodeCommit repository AWS.
- Docker è un insieme di prodotti Platform as a Service (PaaS) che utilizzano la virtualizzazione a livello di sistema operativo per fornire software in container. Questo modello utilizza Docker per creare e testare le immagini dei container localmente.
- cfn-lint e cfn-nag sono strumenti open source che ti aiutano a esaminare CloudFormation gli stack per eventuali errori e problemi di sicurezza.

Archivio di codici

Il codice per questo modello è disponibile nell'archivio GitHub Global Blue/Green deployments in multiple region and accounts.

Epiche

Impostazione delle variabili di ambiente

Operazione	Description	Competenze richieste
Esporta le variabili di ambiente per la distribuzione CloudFormation in stack.	Definisci le variabili di ambiente che verranno utilizzate come input per gli CloudFormation stack più avanti in questo schema. 1. Aggiorna i nomi dei bucket che hai creato nei tre account e regioni come spiegato in precedenza nella sezione Prerequisiti:	AWS DevOps

Operazione	Description	Competenze richieste
	``` export S3BUCKETN AMETESTACCOUNT1=<S 3BUCKETACCOUNT1> export S3BUCKETN AMETESTACCOUNT2=<S 3BUCKETACCOUNT2> export S3BUCKETN AMETESTACCOUNT3=<S 3BUCKETACCOUNT3> ``` 2. Definite una stringa casuale per creare bucket di artefatti , poiché i nomi dei bucket devono essere univoci a livello globale: ``` export BUCKETSTA RTNAME=ecs-codepip eline-artifacts-19 992 ``` 3. Definisci ed esporta l'account e le regioni: IDs ``` export TOOLSACCO UNT=<TOOLSACCOUNT> export CODECOMMI TACCOUNT=<CODECOMM ITACCOUNT> export CODECOMMI TREGION=ap-south-1 export CODECOMMI TREPONAME=Poc export TESTACCOU NT1=<TESTACCOUNT1> export TESTACCOU NT2=<TESTACCOUNT2> export TESTACCOU NT3=<TESTACCOUNT3> 	

Operazione	Description	Competenze richieste

	`export TESTACCOU NT1REGION=ap-south -1`
	`export TESTACCOU NT2REGION=eu-centr al-1`
	`export TESTACCOU NT3REGION=us-east-1`
	`export TOOLSACCO UNTREGION=ap-south -1`
	`export ECRREPOSI TORYNAME=web`	

Package e distribuzione degli CloudFormation stack per l'infrastruttura

Operazione	Description	Competenze richieste

Clonare il repository.	Clona il repository di esempio in un nuovo repository nella tua sede di lavoro:
	`##In work location`
	`git clone https://g ithub.com/aws-samp les/ecs-blue-green -global-deployment- with-multiregion-cmk- codepipeline.git`
		AWS DevOps

Package delle risorse di Cloudformation.	In questo passaggio, impacchetterai gli artefatti locali a cui i CloudFormation modelli fanno riferimento per creare le risorse di infrastru ttura necessarie per servizi come Amazon Virtual Private	AWS DevOps

Operazione	Description	Competenze richieste

	Cloud (Amazon VPC) e Application Load Balancer.
	
	I modelli sono disponibili nella `Infra` cartella del repository del codice.
	
	`##In TestAccount1## aws cloudformation package \ --template-file mainInfraStack.yaml \ --s3-bucket $S3BUCKETNAMETESTACCOUNT1 \ --s3-prefix infraStack \ --region $TESTACCOUNT1REGION \ --output-template-file infrastructure_${TESTACCOUNT1}.template`
	
	`##In TestAccount2## aws cloudformation package \ --template-file mainInfraStack.yaml \ --s3-bucket $S3BUCKETNAMETESTACCOUNT2 \ --s3-prefix infraStack \ --region $TESTACCOUNT2REGION \ --output-template-file infrastructure_`	

Operazione	Description	Competenze richieste

	`${TESTACCOUNT2}.template`
	
	`##In TestAccount3##`
	`aws cloudformation package \`
	`--template-file mainInfraStack.yaml \`
	`--s3-bucket $S3BUCKETNAMETESTACCOUNT3 \`
	`--s3-prefix infraStack \`
	`--region $TESTACCOUNT3REGION \`
	`--output-template-file infrastructure_${TESTACCOUNT3}.template`	

Operazione	Description	Competenze richieste

Convalida i modelli di pacchetto.	Convalida i modelli di pacchetto:
	
	`aws cloudformation validate-template \ --template-body file://infrastructure_${TESTACCOUNT1}.template`
	
	`aws cloudformation validate-template \ --template-body file://infrastructure_${TESTACCOUNT2}.template`
	
	`aws cloudformation validate-template \ --template-body file://infrastructure_${TESTACCOUNT3}.template`
		AWS DevOps

Operazione	Description	Competenze richieste

Distribuisci i file del pacchetto negli account dei carichi di lavoro,	1. Aggiorna i valori segnapost o e i nomi degli account nello `nfraParam eters.json` script i in base alla tua configura zione.
	2. Distribuisci i modelli di pacchetto nei tuoi tre account di carico di lavoro.
	
	`##In TestAccount1## aws cloudformation deploy \ --template-file infrastructure_${T ESTACCOUNT1}.templ ate \ --stack-name mainInfrastack \ --parameter- overrides file://in fraParameters.json \ --region $TESTACCO UNT1REGION \ --capabilities CAPABILITY_IAM CAPABILITY_NAMED_I AM`
	
	`##In TestAccount2## aws cloudformation deploy \ --template-file infrastructure_${T ESTACCOUNT2}.templ ate \ --stack-name mainInfrastack \`
		AWS DevOps

Operazione	Description	Competenze richieste
	--parameter-overrides file://infraParameters.json \ --region $TESTACCOUNT2REGION \ --capabilities CAPABILITY_IAM CAPABILITY_NAMED_IAM ##In TestAccount3## aws cloudformation deploy \ --template-file infrastructure_${TESTACCOUNT3}.template \ --stack-name mainInfrastack \ --parameter-overrides file://infraParameters.json \ --region $TESTACCOUNT3REGION \ --capabilities CAPABILITY_IAM CAPABILITY_NAMED_IAM	

Invia un'immagine di esempio e ridimensiona Amazon ECS

Operazione	Description	Competenze richieste
Invia un'immagine di esempio al repository Amazon ECR.	Invia un'immagine di esempio (NGINX) al repository Amazon Elastic Container Registry (Amazon ECR) `web` denominato (come impostato	AWS DevOps

Operazione	Description	Competenze richieste

	nei parametri). Puoi personalizzare l'immagine come richiesto.
	
	Per accedere e impostare le credenziali per inviare un'immagine ad Amazon ECR, segui le istruzioni nella documentazione di Amazon ECR.
	
	I comandi sono:
	
	`docker pull nginx`
	`docker images`
	`docker tag <imageid> aws_account_id.dkr.ecr.region.amazonaws.com/<web>:latest`
	`docker push <aws_account_id>.dkr.ecr.<region>.amazonaws.com/<web>:tag`	

Operazione	Description	Competenze richieste
Ridimensiona Amazon ECS e verifica l'accesso.	1. Scala Amazon ECS per creare due repliche: `aws ecs update-service --cluster QA-Cluster --service Poc-Service --desired-count 2` dove `Poc-Service` si riferisce alla tua applicazione di esempio. 2. Verifica che i servizi siano accessibili da Application Load Balancer utilizzando un nome di dominio completo (FQDN) o DNS da un browser o utilizzando il comando curl.	AWS DevOps

Configura servizi e risorse di codice

Operazione	Description	Competenze richieste
Crea un CodeCommit repository nell'account degli strumenti.	Crea un CodeCommit repository nell'account degli strumenti utilizzando il `codecommit.yaml` modello, che si trova nella `code` cartella del GitHub repository. È necessario creare questo repository solo nella singola regione in cui si prevede di sviluppare il codice.	AWS DevOps

Operazione	Description	Competenze richieste
	`aws cloudformation deploy --stack-name codecommitrepoStack --parameter-overrides CodeCommitReponame=$CODECOMMITREPONAME \ ToolsAccount=$TOOLSACCOUNT --template-file codecommit.yaml --region $TOOLSACCOUNTREGION \ --capabilities CAPABILITY_NAMED_IAM`	

Operazione	Description	Competenze richieste
Crea un bucket S3 per gestire gli artefatti generati da. CodePipeline	Crea un bucket S3 per gestire gli artefatti generati CodePipeline utilizzando il `pre-reqs-bucket.yaml` modello, che si trova nella cartella del repository. `code` GitHub Gli stack devono essere distribuiti in tutti e tre gli account e le regioni per carichi di lavoro (test) e strumenti. `aws cloudformation deploy --stack-name pre-reqs-artifacts-bucket --parameter-overrides BucketStartName=$BUCKETSTARTNAME \ TestAccount1=$TESTACCOUNT1 TestAccount2=$TESTACCOUNT2 \ TestAccount3=$TESTACCOUNT3 CodeCommitAccount=$CODECOMMITACCOUNT ToolsAccount=$TOOLSACCOUNT \ --template-file pre-reqs_bucket.yaml --region $TESTACCOUNT1REGION --capabilities CAPABILITY_NAMED_IAM`  `aws cloudformation deploy --stack-name pre-reqs-artifacts-bucket --parameter-overrides BucketSta`	AWS DevOps

Operazione	Description	Competenze richieste
	rtName=$BUCKETSTAR TNAME \ TestAccount1=$TE STACCOUNT1 TestAccou nt2=$TESTACCOUNT2 \ TestAccount3=$TE STACCOUNT3 CodeCommi tAccount=$CODECOMM ITACCOUNT ToolsAcco unt=$TOOLSACCOUNT \ --template-file pre- reqs_bucket.yaml --region $TESTACCO UNT2REGION --capabil ities CAPABILIT Y_NAMED_IAM aws cloudformation deploy --stack-name pre-reqs-artifacts -bucket --parameter- overrides BucketSta rtName=$BUCKETSTAR TNAME \ TestAccount1=$TE STACCOUNT1 TestAccou nt2=$TESTACCOUNT2 \ TestAccount3=$TE STACCOUNT3 CodeCommi tAccount=$CODECOMM ITACCOUNT ToolsAcco unt=$TOOLSACCOUNT \ --template-file pre- reqs_bucket.yaml --region $TESTACCO UNT3REGION --capabil ities CAPABILIT Y_NAMED_IAM aws cloudformation deploy --stack-name pre-reqs-artifacts	

Operazione	Description	Competenze richieste
	`-bucket --parameter-overrides BucketStartName=$BUCKETSTARTNAME \ TestAccount1=$TESTACCOUNT1 TestAccount2=$TESTACCOUNT2 \ TestAccount3=$TESTACCOUNT3 CodeCommitAccount=$CODECOMMITACCOUNT ToolsAccount=$TOOLSACCOUNT \ --template-file pre-reqs_bucket.yaml --region $TOOLSACCOUNTREGION --capabilities CAPABILITY_NAMED_IAM`	

Operazione	Description	Competenze richieste
Configura una chiave KMS multiregionale.	1. Crea una chiave KMS multiregionale con chiavi primarie e di replica che utilizzerai. CodePipeline Nel nostro esempio, `ToolsAccount1region - ap-south-1` sarà la regione principale. `aws cloudformation deploy --stack-name ecs-codepipeline-pre-reqs-KMS \ --template-file pre-reqs_KMS.yaml --parameter-overrides \ TestAccount1=$TESTACCOUNT1 TestAccount2=$TESTACCOUNT2 \ TestAccount3=$TESTACCOUNT3 CodeCommitAccount=$CODECOMMITACCOUNT ToolsAccount=$TOOLSACCOUNT --region $TOOLSACCOUNTREGION` 2. Imposta le variabili CMKARN da passare ai CodeBuild progetti. I valori sono disponibili nell'output dello stack di modelli ecs-codepipeline-pre-reqs -KMS (l'ID della chiave sarà lo stesso in tutte le regioni e inizia con). `mrk-` In alternativa, puoi ottenere i valori	AWS DevOps

Operazione	Description	Competenze richieste

	CMKARN dall'account degli strumenti. Esportali in tutte le sessioni dell'account:
	
	`export CMKARN1=arn:aws:kms:ap-south-1:<TOOLSACCOUNTID>:key/mrk-xxx`
	`export CMKARN2=arn:aws:kms:eu-central-1:<TOOLSACCOUNTID>:key/mrk-xxx`
	`export CMKARN3=arn:aws:kms:us-east-1:<TOOLSACCOUNTID>:key/mrk-xxx`
	`export CMARNTOOLS=arn:aws:kms:ap-south-1:<TOOLSACCOUNTID>:key/mrk-xxx`	

Operazione	Description	Competenze richieste

Configura il CodeBuild progetto nell'account degli strumenti.	1. Utilizza il `codebuild_IAM.yaml` modello dalla `code` cartella del GitHub repository per configurare AWS Identity and Access Management (IAM) per AWS CodeBuild in una singola regione nell'account degli strumenti:
	`#In ToolsAccount`
	`aws cloudformation deploy --stack-name ecs-codebuild-iam \`
	`--template-file codebuild_IAM.yaml --region $TOOLSACCOUNTREGION \`
	`--capabilities CAPABILITY_NAMED_IAM`
	2. Usa il `codebuild.yaml` modello CodeBuild per configurare il tuo progetto di build. Distribuisci questo modello in tutte e tre le regioni come segue:
	`aws cloudformation deploy --stack-name ecscodebuildstack --parameter-overrides ToolsAccount=$TOOLSACCOUNT \`
	`CodeCommitRepoName=$CODECOMMITREPONAME ECRRepositoryName=`
		AWS DevOps

Operazione	Description	Competenze richieste
	`$ECRREPOSITORYNAME APPACCOUNTID=$TEST ACCOUNT1 \ TestAccount3=$TE STACCOUNT3 CodeCommi tRegion=$CODECOMMI TREGION CMKARN=$C MKARN1 \ --template-file codebuild.yaml --region $TESTACCO UNT1REGION --capabil ities CAPABILIT Y_NAMED_IAM` `aws cloudformation deploy --stack-name ecscodebuildstack -- parameter-overrides ToolsAccount=$TOOL SACCOUNT \ CodeCommitRepoName= $CODECOMMITREPONAME ECRRepositoryName= $ECRREPOSITORYNAME APPACCOUNTID=$TEST ACCOUNT2 \ TestAccount3=$TE STACCOUNT3 CodeCommi tRegion=$CODECOMMI TREGION CMKARN=$C MKARN2 \ --template-file codebuild.yaml --region $TESTACCO UNT2REGION --capabil ities CAPABILIT Y_NAMED_IAM`  `aws cloudformation deploy --stack-name ecscodebuildstack --`	

Operazione	Description	Competenze richieste
	`parameter-overrides ToolsAccount=$TOOLSACCOUNT \ CodeCommitRepoName=$CODECOMMITREPONAME ECRRepositoryName=$ECRREPOSITORYNAME APPACCOUNTID=$TESTACCOUNT3 \ CodeCommitRegion=$CODECOMMITREGION CMKARN=$CMKARN3 \ --template-file codebuild.yaml --region $TESTACCOUNT3REGION --capabilities CAPABILITY_NAMED_IAM`	

Operazione	Description	Competenze richieste
Configurazione CodeDeploy negli account di carico di lavoro.	Utilizza il `codedeploy.yaml` modello nella `code` cartella del GitHub repository per eseguire la configurazione CodeDeploy in tutti e tre gli account di carico di lavoro. L'output di `mainInfraStack` include Amazon Resource Names (ARNs) del cluster Amazon ECS e il listener Application Load Balancer. **Note** I valori degli stack di infrastruttura vengono già esportati, quindi vengono importati dai modelli di stack. CodeDeploy `##WorkloadAccount1## aws cloudformation deploy --stack-name ecscodedeploystack \ --parameter-overrides ToolsAccount=$TOOLSACCOUNT mainInfrastackname=mainInfrastack \ --template-file codedeploy.yaml --region $TESTACCOUNT1REGION --capabil`	AWS DevOps

Operazione	Description	Competenze richieste
	ities CAPABILIT Y_NAMED_IAM ##WorkloadAccount2## aws cloudformation deploy --stack-name ecscodedeploystack \ --parameter-overrides ToolsAccount=$TOOL SACCOUNT mainInfra stackname=mainInfr astack \ --template-file codedeploy.yaml --region $TESTACCO UNT2REGION --capabil ities CAPABILIT Y_NAMED_IAM ##WorkloadAccount3## aws cloudformation deploy --stack-name ecscodedeploystack \ --parameter-overrides ToolsAccount=$TOOL SACCOUNT mainInfra stackname=mainInfr astack \ --template-file codedeploy.yaml --region $TESTACCO UNT3REGION --capabil ities CAPABILIT Y_NAMED_IAM	

Configura CodePipeline nell'account degli strumenti

Operazione	Description	Competenze richieste
Crea una pipeline di codice nell'account degli strumenti.	Nell'account degli strumenti, esegui il comando: `aws cloudformation deploy --stack-name ecscodepipelinestack --parameter-overrides \ TestAccount1=$TESTACCOUNT1 TestAccount1Region=$TESTACCOUNT1REGION \ TestAccount2=$TESTACCOUNT2 TestAccount2Region=$TESTACCOUNT2REGION \ TestAccount3=$TESTACCOUNT3 TestAccount3Region=$TESTACCOUNT3REGION \ CMKARNTools=$CMKTROOLSARN CMKARN1=$CMKARN1 CMKARN2=$CMKARN2 CMKARN3=$CMKARN3 \ CodeCommitRepoName=$CODECOMMITREPONAME BucketStartName=$BUCKETSTARTNAME \ --template-file codepipeline.yaml --capabilities CAPABILITY_NAMED_IAM`	AWS DevOps
Fornisci accesso CodePipeline e CodeBuild ruoli nella policy	1. Fornisci l'accesso CodePipeline e CodeBuild	AWS DevOps

Operazione	Description	Competenze richieste
chiave di AWS KMS e nella policy del bucket S3.	i ruoli nella policy chiave di AWS KMS: `aws cloudformation deploy --stack-name ecs-codepipeline-pre-reqs-KMS \ --template-file pre-reqs_KMS.yaml --parameter-overrides \ CodeBuildCondition=true TestAccount1=$TESTACCOUNT1 TestAccount2=$TESTACCOUNT2 \ TestAccount3=$TESTACCOUNT3 CodeCommitAccount=$CODECOMMITACCOUNT ToolsAccount=$TOOLSACCOUNT --region $TOOLSACCOUNTREGION`  2. Aggiorna la policy sui bucket S3 per consentire l'accesso a e ruoli: CodePipeline CodeDeploy  `aws cloudformation deploy --stack-name pre-reqs-artifacts-bucket --parameter-overrides BucketStartName=$BUCKETSTARTNAME \ PutS3BucketPolicy=true TestAccount1=$TESTACCOUNT1`	

Operazione	Description	Competenze richieste
	TestAccount2=$TEST ACCOUNT2 \ TestAccount3=$TE STACCOUNT3 CodeCommi tAccount=$CODECOMM ITACCOUNT ToolsAcco unt=$TOOLSACCOUNT \ --template-file pre- reqs_bucket.yaml --region $TESTACCO UNT1REGION --capabil ities CAPABILIT Y_NAMED_IAM aws cloudformation deploy --stack-name pre-reqs-artifacts -bucket --parameter- overrides BucketSta rtName=$BUCKETSTAR TNAME \ PutS3BucketPolic y=true TestAccou nt1=$TESTACCOUNT1 TestAccount2=$TEST ACCOUNT2 \ TestAccount3=$TE STACCOUNT3 CodeCommi tAccount=$CODECOMM ITACCOUNT ToolsAcco unt=$TOOLSACCOUNT \ --template-file pre- reqs_bucket.yaml --region $TESTACCO UNT2REGION --capabil ities CAPABILIT Y_NAMED_IAM aws cloudformation deploy --stack-name pre-reqs-artifacts -bucket --parameter-	

Operazione	Description	Competenze richieste
	overrides BucketStartName=$BUCKETSTARTNAME \ PutS3BucketPolicy=true TestAccount1=$TESTACCOUNT1 TestAccount2=$TESTACCOUNT2 \ TestAccount3=$TESTACCOUNT3 CodeCommitAccount=$CODECOMMITACCOUNT ToolsAccount=$TOOLSACCOUNT \ --template-file pre-reqs_bucket.yaml --region $TESTACCOUNT3REGION --capabilities CAPABILITY_NAMED_IAM aws cloudformation deploy --stack-name pre-reqs-artifacts-bucket --parameter-overrides BucketStartName=$BUCKETSTARTNAME \ PutS3BucketPolicy=true TestAccount1=$TESTACCOUNT1 TestAccount2=$TESTACCOUNT2 \ TestAccount3=$TESTACCOUNT3 CodeCommitAccount=$CODECOMMITACCOUNT ToolsAccount=$TOOLSACCOUNT \ --template-file pre-reqs_bucket.yaml --region $TOOLSACCOUNTREGION --capabil	

Operazione	Description	Competenze richieste
	`ities CAPABILIT Y_NAMED_IAM`	

Chiama e testa la pipeline

Operazione	Description	Competenze richieste
Invia le modifiche al CodeCommit repository.	1. Clona il CodeCommit repository creato nel `codecommitrepoStack` by utilizzando il `git clone` comando, come descritto nella documentazione CodeCommit AWS. 2. Aggiorna gli artefatti di input con i dettagli richiesti: • File JSON: aggiorna `AccountID` il file in tre punti del file. Rinomina i tre file per includere l'account. IDs • File YAML: aggiorna l'ARN e la versione della definizione dell'attività. Rinomina i tre file per includere l'account. IDs 3. Modifica il `index.html` file per apportare alcune modifiche minori alla home page. 4. Copia i seguenti file nel repository ed esegui il commit:	

Operazione	Description	Competenze richieste
	index.html Dockerfile buildspec.yaml appspec_<accountid>.yaml (3 files - one per account) taskdef<accountid>.json (3 files - one per account) 5. Avvia o riavvia la pipeline e verifica i risultati. 6. Accedi al servizio dall'Application Load Balancer utilizzando un FQDN o DNS e verifica che gli aggiornamenti siano stati distribuiti.	

Eliminazione

Operazione	Description	Competenze richieste
Pulisci tutte le risorse distribuite.	1. Ridimensiona Amazon ECS a zero istanze: aws ecs update-service --cluster QA-Cluster --service Poc-Service --desired-count 0 2. Elimina gli CloudFormation stack in ogni account e regione: ##In Tools Account##	

Operazione	Description	Competenze richieste

	`aws cloudformation delete-stack --stack-name ecscodepipelinestack --region $TOOLSACCOUNTREGION`
	`aws cloudformation delete-stack --stack-name ecscodebuildstack --region $TESTACCOUNT1REGION`
	`aws cloudformation delete-stack --stack-name ecscodebuildstack --region $TESTACCOUNT2REGION`
	`aws cloudformation delete-stack --stack-name ecscodebuildstack --region $TESTACCOUNT3REGION`
	`aws cloudformation delete-stack --stack-name ecs-codepipeline-pre-reqs-KMS --region $TOOLSACCOUNTREGION`
	`aws cloudformation delete-stack --stack-name codecommitrepoStack --region $TOOLSACCOUNTREGION`
	`aws cloudformation delete-stack --stack-name pre-reqs-artifacts-bucket --region $TESTACCOUNT1REGION`
	`aws cloudformation delete-stack --stack-name pre-reqs-artifacts-bucket`	

Operazione	Description	Competenze richieste
	--region $TESTACCO UNT2REGION aws cloudformation delete-stack -- stack-name pre-reqs- artifacts-bucket --region $TESTACCO UNT3REGION aws cloudformation delete-stack -- stack-name pre-reqs- artifacts-bucket --region $TOOLSACC OUNTREGION aws cloudformation delete-stack -- stack-name ecs-codeb uild-iam --region $TOOLSACCOUNTREGION ##NOTE: Artifact buckets will not get deleted if there are artifacts so it has to be emptied manually before deleting.## ##In Workload / Test Accounts## ##Account:1## aws cloudformation delete-stack -- stack-name ecscodede ploystack --region $TESTACCOUNT1REGION aws cloudformation delete-stack -- stack-name mainInfra stack --region $TESTACCOUNT1REGION	

Operazione	Description	Competenze richieste
	``` ##Account:2## aws cloudformation delete-stack -- stack-name ecscodede ploystack --region $TESTACCOUNT2REGION aws cloudformation delete-stack -- stack-name mainInfra stack --region $TESTACCOUNT2REGION ##Account:3## aws cloudformation delete-stack -- stack-name ecscodede ploystack --region $TESTACCOUNT3REGION aws cloudformation delete-stack -- stack-name mainInfra stack --region $TESTACCOUNT3REGION ##NOTE: Amazon ECR (web) will not get deleted if the registry still includes images. It can be manually cleaned up if not required. ```	

Risoluzione dei problemi

Problema	Soluzione
Le modifiche che hai eseguito nel repository non vengono distribuite.	• Controlla la presenza di errori nei CodeBuild log nell'azione di compilazione di Docker. Per ulteriori informazioni, consulta la documentazione relativa ad CodeBuild .

Problema	Soluzione
	• Controlla la CodeDeploy distribuzione per eventuali problemi di distribuzione di Amazon ECS.

Risorse correlate

- Inviare un'immagine Docker (documentazione Amazon ECR)
- Connect a un CodeCommit repository AWS (CodeCommit documentazione AWS)
- Risoluzione dei problemi AWS CodeBuild (CodeBuild documentazione AWS)

Monitora i repository Amazon ECR per le autorizzazioni wildcard utilizzando AWS e AWS Config CloudFormation

Vikrant Telkar, Wassim Benhallam e Sajid Momin, Amazon Web Services

Riepilogo

Sul cloud Amazon Web Services (AWS), Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito che supporta repository privati con autorizzazioni basate su risorse utilizzando AWS Identity and Access Management (IAM).

IAM supporta il carattere jolly * "" sia negli attributi resource che action, il che semplifica la scelta automatica di più elementi corrispondenti. Nel tuo ambiente di test, puoi consentire a tutti gli utenti AWS autenticati di accedere a un repository Amazon ECR utilizzando l'autorizzazione `ecr:*` wildcard in un elemento principale della dichiarazione sulla politica del repository. L'autorizzazione `ecr:*` wildcard può essere utile per lo sviluppo e il test in account di sviluppo che non possono accedere ai dati di produzione.

Tuttavia, è necessario assicurarsi che l'autorizzazione `ecr:*` con i caratteri jolly non venga utilizzata negli ambienti di produzione perché può causare gravi vulnerabilità di sicurezza. L'approccio di questo modello ti aiuta a identificare i repository Amazon ECR che contengono l'autorizzazione `ecr:*` wildcard nelle dichiarazioni sulle politiche relative ai repository. Il modello fornisce i passaggi e un CloudFormation modello AWS per creare una regola personalizzata in AWS Config. Una funzione AWS Lambda monitora quindi le dichiarazioni sulla policy del repository Amazon ECR per le autorizzazioni wildcard. `ecr:*` Se rileva dichiarazioni di policy relative ai repository non conformi, Lambda notifica ad AWS Config l'invio di un evento ad Amazon EventBridge e EventBridge quindi avvia un argomento su Amazon Simple Notification Service (Amazon SNS). L'argomento SNS ti notifica via e-mail le dichiarazioni sulle politiche relative ai repository non conformi.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- AWS Command Line Interface (AWS CLI), installata e configurata. Per ulteriori informazioni su questo argomento, consulta Installazione, aggiornamento e disinstallazione dell'interfaccia a riga di comando di AWS nella documentazione dell'interfaccia a riga di comando di AWS.
- Un repository Amazon ECR esistente con una dichiarazione di policy allegata, installato e configurato nel tuo ambiente di test. Per ulteriori informazioni su questo argomento, consulta

Creazione di un repository privato e Impostazione di una dichiarazione sulla politica del repository nella documentazione di Amazon ECR.

- AWS Config, configurato nella tua regione AWS preferita. Per ulteriori informazioni su questo argomento, consulta Getting started with AWS Config nella documentazione di AWS Config.
- Il `aws-config-cloudformation.template` file (allegato), scaricato sul computer locale.

Limitazioni

- La soluzione di questo modello è regionale e le risorse devono essere create nella stessa regione.

Architecture

Il diagramma seguente mostra come AWS Config valuta le dichiarazioni sulle policy dei repository Amazon ECR.

Il diagramma mostra il flusso di lavoro seguente:

1. AWS Config avvia una regola personalizzata.
2. La regola personalizzata richiama una funzione Lambda per valutare la conformità delle dichiarazioni politiche del repository Amazon ECR. La funzione Lambda identifica quindi le dichiarazioni di policy del repository non conformi.
3. La funzione Lambda invia lo stato di non conformità ad AWS Config.
4. AWS Config invia un evento a. EventBridge
5. EventBridge pubblica le notifiche di non conformità su un argomento SNS.
6. Amazon SNS invia un avviso e-mail a te o a un utente autorizzato.

Automazione e scalabilità

La soluzione di questo modello è in grado di monitorare un numero qualsiasi di dichiarazioni sulla politica dei repository Amazon ECR, ma tutte le risorse che desideri valutare devono essere create nella stessa regione.

Tools (Strumenti)

- AWS CloudFormation: AWS ti CloudFormation aiuta a modellare e configurare le tue risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle per tutto il loro ciclo di vita. Puoi utilizzare un modello per descrivere le tue risorse e le loro dipendenze e lanciarle e configurarle insieme come uno stack, invece di gestire le risorse singolarmente. Puoi gestire e fornire stack su più account AWS e regioni AWS.
- AWS Config: AWS Config fornisce una visualizzazione dettagliata della configurazione delle risorse AWS nel tuo account AWS. Questo include le relazioni tra le risorse e la maniera in cui sono state configurate in passato, in modo che tu possa vedere come le configurazioni e le relazioni cambiano nel corso del tempo.
- Amazon ECR — Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito da AWS sicuro, scalabile e affidabile. Amazon ECR supporta i repository privati con autorizzazioni basate sulle risorse utilizzando IAM.
- Amazon EventBridge: Amazon EventBridge è un servizio di bus eventi senza server che puoi utilizzare per connettere le tue applicazioni con dati provenienti da una varietà di fonti. EventBridge fornisce un flusso di dati in tempo reale dalle tue applicazioni, dalle applicazioni SaaS (SaaS) e dai servizi AWS a target come funzioni AWS Lambda, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altri account.
- AWS Lambda: AWS Lambda è un servizio di elaborazione che supporta l'esecuzione di codice senza effettuare il provisioning o la gestione di server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo. Verrà addebitato soltanto il tempo di calcolo consumato e non verrà addebitato alcun costo quando il codice non è in esecuzione.
- Amazon SNS — Amazon Simple Notification Service (Amazon SNS) coordina e gestisce la consegna o l'invio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail. I sottoscrittori ricevono tutti gli stessi messaggi pubblicati sugli argomenti ai quali sono hanno effettuato la sottoscrizione.

Codice

Il codice per questo pattern è disponibile nel `aws-config-cloudformation.template` file (allegato).

Epiche

Crea lo CloudFormation stack AWS

Operazione	Description	Competenze richieste
Crea lo CloudFormation stack AWS.	Crea uno CloudFormation stack AWS eseguendo il seguente comando nell'interfaccia a riga di comando di AWS: `$ aws cloudformation create-stack --stack-name=AWSConfigECR \ --template-body file://aws-config-cloudformation.template \ --parameters ParameterKey=<email>,ParameterValue=<myemail@example.com> \ --capabilities CAPABILITY_NAMED_IAM`	AWS DevOps

Prova la regola personalizzata di AWS Config

Operazione	Description	Competenze richieste
Prova la regola personalizzata di AWS Config.	1. Accedi alla Console di gestione AWS, apri la console AWS Config e scegli Risorse. 2. Nella pagina Inventario delle risorse, puoi filtrare per categoria di risorse,	AWS DevOps

Operazione	Description	Competenze richieste
	tipo di risorsa e stato di conformità. 3. Un repository Amazon ECR che contiene lo `ecr:*` è `NON-COMPLIANT?` e un repository Amazon ECR che non contiene lo è. `ecr:*` `COMPLIANT` 4. L'indirizzo e-mail sottoscritto all'argomento SNS riceve notifiche se un repository Amazon ECR contiene dichiarazioni di policy non conformi.	

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Ottimizza le implementazioni serverless multi-account utilizzando i flussi di lavoro e Actions AWS CDK GitHub

Sarat Chandra Pothula e VAMSI KRISHNA SUNKAVALLI, Amazon Web Services

Riepilogo

Organizations che implementano un'infrastruttura serverless su più Account AWS ambienti spesso incontrano sfide come la duplicazione del codice, i processi manuali e le pratiche non coerenti. La soluzione di questo modello mostra come utilizzare i flussi di lavoro riutilizzabili di Go and GitHub Actions per semplificare la AWS Cloud Development Kit (AWS CDK) gestione dell'infrastruttura serverless con più account. Questa soluzione dimostra come definire le risorse cloud come codice, implementare processi di integration/continuous distribuzione continua standardizzati (CI/CD) e creare componenti modulari e riutilizzabili.

Utilizzando questi strumenti, le organizzazioni possono gestire in modo efficiente le risorse tra account, implementare pipeline di distribuzione coerenti e semplificare architetture serverless complesse. L'approccio migliora anche la sicurezza e la conformità applicando pratiche standardizzate di utilizzo Account AWS, migliorando in ultima analisi la produttività e riducendo gli errori nello sviluppo e nella distribuzione di applicazioni serverless.

Prerequisiti e limitazioni

Prerequisiti

- Un Account AWS attivo.
- AWS Identity and Access Management I ruoli e le autorizzazioni (IAM) sono disponibili per il processo di distribuzione. Ciò include le autorizzazioni per accedere ai repository Amazon Elastic Container Registry (Amazon ECR), creare AWS Lambda funzioni e qualsiasi altra risorsa richiesta nell'ambito della destinazione. Account AWS
- AWS Command Line Interface (AWS CLI) versione 2.9.11 o successiva, installata e configurata.
- AWS Cloud Development Kit (AWS CDK) versione 2.114.1 o successiva, installata e avviata.
- Go 1.22 o versione successiva, installata.
- Docker 24.0.6 o versione successiva, installato.

Limitazioni

- Compatibilità linguistica: Go è un linguaggio popolare per le applicazioni serverless. Tuttavia, oltre a Go, AWS CDK supporta altri linguaggi di programmazione, tra cui C#, Java, Python e. TypeScript Se la tua organizzazione dispone di basi di codice esistenti o ha esperienza in altri linguaggi, potrebbe essere necessario adattare o imparare a usare Go per utilizzare appieno la soluzione descritta nel modello.
- Curva di apprendimento: l'adozione di flussi di lavoro AWS CDK, Go (se è una novità per l'organizzazione) e GitHub riutilizzabili potrebbe comportare una curva di apprendimento per sviluppatori e team. DevOps Potrebbero essere necessarie formazione e documentazione per garantire un'adozione agevole e un uso efficace di queste tecnologie.

Architecture

Il diagramma seguente mostra i componenti del flusso di lavoro e dell'architettura di questo modello.

Questa soluzione esegue i seguenti passaggi:

1. Lo sviluppatore clona il repository, crea un nuovo ramo e apporta modifiche al codice dell'applicazione nel proprio ambiente locale.
2. Lo sviluppatore esegue queste modifiche e inserisce il nuovo ramo nel repository. GitHub
3. Lo sviluppatore crea una pull request nel GitHub repository, proponendo di unire la propria funzionalità o il nuovo ramo di funzionalità nel ramo principale.
4. Questa pull request attiva il flusso di lavoro Continuous Integration (CI) Actions. GitHub I flussi di lavoro CI e di distribuzione continua (CD) in questo modello utilizzano flussi di lavoro riutilizzabili, ovvero modelli modulari predefiniti che possono essere condivisi ed eseguiti su diversi progetti o repository. I flussi di lavoro riutilizzabili promuovono la standardizzazione e l'efficienza dei processi. CI/CD
5. Il flusso di lavoro CI configura l'ambiente necessario, genera un tag Docker per l'immagine e crea l'immagine Docker utilizzando il codice dell'applicazione.
6. Il flusso di lavoro CI si autentica AWS utilizzando il ruolo centrale OIDC. Account AWS GitHub Per i flussi di lavoro CI, il ruolo Account AWS GitHub OIDC centrale utilizza AWS Security Token Service () per ottenere credenziali temporanee.AWS STS Queste credenziali consentono al ruolo di creare e inviare immagini Docker all'archivio Amazon ECR della centrale. Account AWS
7. Il flusso di lavoro CI trasferisce l'immagine Docker integrata in Amazon ECR.
8. Il flusso di lavoro CI memorizza il tag dell'immagine nel Systems Manager Parameter Store.

9. Una volta completato correttamente il flusso di lavoro CI, viene emesso il tag di immagine Docker.
10. Quando attiva il flusso di lavoro del CD, lo sviluppatore inserisce manualmente il tag di immagine dell'immagine Docker che desidera distribuire. Questo tag immagine corrisponde al tag generato e inviato ad Amazon ECR durante il flusso di lavoro CI.
11. Lo sviluppatore attiva manualmente il flusso di lavoro del CD, che utilizza il flusso di lavoro riutilizzabile del CD.
12. Il flusso di lavoro del CD si autentica AWS utilizzando il ruolo centrale OIDC. Account AWS GitHub Per il flusso di lavoro del CD, AWS STS viene utilizzato innanzitutto per assumere il ruolo centrale dell'OIDC. Account AWS GitHub Quindi, questo ruolo assume i ruoli di bootstrap CDK per le distribuzioni degli account di destinazione.
13. Il flusso di lavoro del CD utilizza i modelli per sintetizzare. AWS CDK AWS CloudFormation
14. Il flusso di lavoro CD distribuisce l'applicazione sulla destinazione Account AWS utilizzando CDK deploy, utilizzando il tag immagine specificato manualmente per la funzione Lambda.

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che consente di definire e fornire Cloud AWS l'infrastruttura nel codice.
- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita tra Account AWS e. Regioni AWS CloudFormation è parte integrante del processo di AWS CDK implementazione. Il CDK sintetizza i CloudFormation modelli e li utilizza CloudFormation per creare o aggiornare le risorse nell'ambiente. AWS
- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS Systems Manager Parameter Store fornisce uno storage sicuro e gerarchico per la gestione dei dati di configurazione e la gestione dei segreti.

Altri strumenti

- Docker è un insieme di prodotti Platform as a Service (PaaS) che utilizzano la virtualizzazione a livello di sistema operativo per fornire software in container.
- GitHub Actions è una piattaforma di integrazione e distribuzione continua (CI/CD) strettamente integrata con i repository. GitHub Puoi utilizzare GitHub Actions per automatizzare la pipeline di compilazione, test e distribuzione.
- Go è un linguaggio di programmazione open source supportato da Google.

Archivio di codici

Il codice per questo pattern è disponibile nel cicd-github-actions repository GitHub aws-cdk-golang-serverless-.

Best practice

- Design modulare: organizza il AWS CDK codice in costrutti o pile modulari e riutilizzabili, promuovendo il riutilizzo e la manutenibilità del codice su più account e progetti.
- Separazione delle preoccupazioni: separa il codice dell'infrastruttura dal codice dell'applicazione, consentendo l'implementazione e la gestione indipendenti di ogni componente.
- Controllo delle versioni e immutabilità: considera la tua infrastruttura come codice (IaC) e usa Git per il controllo delle versioni. Abbraccia i principi dell'infrastruttura immutabile creando nuove risorse invece di modificare quelle esistenti.
- Test e convalida: implementa strategie di test complete, tra cui test unitari, test di integrazione e end-to-end test, per supportare la correttezza e l'affidabilità del codice e delle implementazioni. AWS CDK
- Sicurezza e conformità: segui le migliori pratiche AWS di sicurezza, come l'accesso con privilegi minimi, la comunicazione sicura e la crittografia dei dati. Implementa controlli di conformità e meccanismi di verifica per garantire il rispetto delle politiche organizzative e dei requisiti normativi. Implementa le migliori pratiche di sicurezza per le immagini dei container, come la scansione delle vulnerabilità, l'applicazione della firma delle immagini e il rispetto dei requisiti di conformità per la tua organizzazione.
- Monitoraggio e registrazione: configura meccanismi di monitoraggio e registrazione per monitorare lo stato e le prestazioni delle applicazioni e dell'infrastruttura serverless. Servizi AWS Utilizzalo come Amazon CloudWatch AWS CloudTrail e AWS X-Ray per scopi di monitoraggio e controllo.

- Automazione e CI/CD: utilizza flussi di lavoro GitHub riutilizzabili e altri CI/CD strumenti per automatizzare i processi di creazione, test e distribuzione, il che può aiutare a supportare implementazioni coerenti e ripetibili su più account.
- Gestione dell'ambiente: gestione di ambienti separati (ad esempio, sviluppo, gestione temporanea e produzione). Implementa strategie per promuovere i cambiamenti tra gli ambienti, garantendo test e convalida adeguati prima delle implementazioni di produzione.
- Documentazione e collaborazione: documenta il codice dell'infrastruttura, i processi di implementazione e le migliori pratiche per facilitare la condivisione delle conoscenze e la collaborazione all'interno del team.
- Ottimizzazione dei costi: implementazione di strategie di monitoraggio e ottimizzazione dei costi, come il corretto dimensionamento delle risorse, l'utilizzo dell'auto-scaling e lo sfruttamento di servizi di ottimizzazione dei AWS costi come e. Budget AWS AWS Cost Explorer
- Disaster recovery e backup: pianifica gli scenari di disaster recovery implementando meccanismi di backup e ripristino per le applicazioni e le risorse dell'infrastruttura serverless.
- Miglioramento continuo: rivedi e aggiorna regolarmente le procedure, gli strumenti e i processi per allinearli alle migliori pratiche, ai consigli sulla sicurezza e ai progressi tecnologici più recenti nell'ecosistema serverless.
- Migliora il livello di sicurezza: AWS PrivateLinkutilizzalo per migliorare il livello di sicurezza del tuo cloud privato virtuale (VPC) configurando gli endpoint VPC di interfaccia per Amazon ECR e Parameter Store. AWS Lambda AWS Systems Manager

Epiche

Configura gli ambienti

Operazione	Description	Competenze richieste
Crea un repository Amazon ECR nella centrale. Account AWS	Per condividere le immagini dei container tra più account Account AWS, devi configura re l'accesso tra più account per Amazon ECR. Innanzitu tto, crea un repository Amazon ECR nella centrale. Account AWS	AWS DevOps

Operazione	Description	Competenze richieste
	Per creare un repository Amazon ECR, esegui il seguente comando: `aws ecr create-repository --repository-name sample-repo` In un'attività successiva, concedi l'accesso pull agli altri Account AWS che devono utilizzare l'immagine del contenitore.	

Operazione	Description	Competenze richieste
Aggiungi autorizzazioni per più account al repository Amazon ECR.	Per aggiungere autorizzazioni per più account al repository Amazon ECR nella centrale Account AWS, esegui il codice seguente: `{ "Version": "2008-10-17", "Statement": [{ "Sid": "LambdaECRImageRetrievalPolicy", "Effect": "Allow", "Principal": { "Service": "lambda.amazonaws.com" }, "Action": ["ecr:BatchGetImage", "ecr:GetDownloadUrlForLayer",], "Condition": { "StringLike": { "aws:sourceArn": "arn:aws:lambda:<Target_Region>:<Target_Account_ID>:function:*" } } }, { "Sid": "new statement", "Effect": "Allow",`	AWS DevOps

Operazione	Description	Competenze richieste
	"Principal": { "AWS": "arn:aws:iam::<Target_Account_ID>:root" }, "Action": ["ecr:BatchGetImage", "ecr:GetDownloadUrlForLayer",], }] }	

Operazione	Description	Competenze richieste
Configura un ruolo per il ruolo GitHub OIDC nella centrale. Account AWS	1. Configura AWS to trust GitHub OIDC come identità federata, che include l'aggiunta del provider GitHub OIDC e la configura zione del ruolo AWS e della policy di fiducia in IAM. Per farlo, segui le istruzioni in Configurazione di OpenID Connect in Amazon Web Services GitHub nella documentazione. 2. Dopo aver creato il ruolo, aggiungi le autorizza zioni necessarie al ruolo. Ad esempio, aggiungi le autorizzazioni per Amazon ECR e AWS Systems Manager Parameter Store. Per ulteriori informazioni, consulta Configurazione di un ruolo per il provider di identità GitHub OIDC nella documentazione IAM.	AWS DevOps

Operazione	Description	Competenze richieste
Avvia l' AWS ambiente nel bersaglio Account AWS.	Configura un ambiente CDK in un ambiente specifico Regione AWS che consenta le distribuzioni tra account da un account centrale Account AWS e applichi i principi del privilegio minimo al ruolo di esecuzione. CloudFormation Per avviare un ambiente, esegui il seguente comando: AWS `cdk bootstrap aws://<Target_Account_ID>/<Target_Region> --trust <Central_Account_ID> --cloudformation-execution-policies arn:aws:iam::aws:policy/<Least_Privilege_Policy>`	AWS DevOps

Operazione	Description	Competenze richieste
Concedi l'accesso al ruolo Account AWS OIDC centrale ai ruoli Account AWS bootstrap di destinazione.	Il bootstrap CDK crea i seguenti ruoli IAM, progettat i per essere assunti dalla centrale Account AWS durante le varie fasi del processo di implementazione CDK: • Ruolo di pubblicazione dei file • Ruolo di pubblicazione di immagini • Ruolo di ricerca • Ruolo di distribuzione Ogni ruolo dispone di autorizzazioni specifich e adattate al suo scopo, secondo il principio del privilegio minimo. Il nome `Target_Account_ID` e `Target_Region` in ogni ruolo aiuta a indicare che questi ruoli sono unici nelle diverse regioni. Account AWS Questo approccio supporta l'identificazione e la gestione chiare in configurazioni con più account e più regioni. `Target Account CDK Bootstrap Roles arn:aws:iam::<Target_Account_ID>:role/cdk-deploy-role-<Ta`	AWS DevOps

Operazione	Description	Competenze richieste
	`rget_Account_ID>-< Target_Region> arn:aws:iam::<Target_ Account_ID>:role/cdk- file-publishing-role- <Target_Account_ID>-< Target_Region> arn:aws:iam::<Target_ Account_ID>:role/cdk- image-publishing-role- <Target_Account_ID>- <Target_Region> arn:aws:iam::<Target _Account_ID>:role/ cdk-lookup-role-<T arget_Account_ID>- <Target_Region>` • Aggiorna la politica di autorizzazione per il ruolo OIDC nell'account centrale per concedergli la possibilità di assumere ruoli nell'acco unt di destinazione. Questa configurazione consente la distribuzione di stack CDK tra diversi. Account AWS Consentendo al ruolo OIDC dell'account centrale di adottare le autorizza zioni necessarie dall'acco unt di destinazione, si crea un ponte sicuro per le implementazioni CDK tra account. Questo approccio mantiene un controllo adeguato degli accessi,	

Operazione	Description	Competenze richieste
	facilitando al contempo la gestione senza interruzioni dell'infrastruttura multi-account. Per aggiornare la politica di autorizzazione per il ruolo OIDC nella centrale Account AWS, utilizza il codice seguente: `{ "Version": "2012-10-17", "Statement": [{ "Effect": "Allow", "Action": "sts:AssumeRole", "Resource": ["arn:aws:iam::<Target_Account_ID>:role/cdk-deploy-role-<Target_Account_ID>-<Target_Region>", "arn:aws:iam::<Target_Account_ID>:role/cdk-file-publishing-role-<Target_Account_ID>-<Target_Region>", "arn:aws:iam::<Target_Account_ID>:role/cdk-image-publishing-`	

Operazione	Description	Competenze richieste
	role-<Target_Account_ID>-<Target_Region>", "arn:aws:iam::<Target_Account_ID>:role/cdk-lookup-role-<Target_Account_ID>-<Target_Region>"] }] }	

Crea l'immagine Docker

Operazione	Description	Competenze richieste
Clona il repository del progetto.	Per clonare il GitHub repository di questo pattern, esegui il seguente comando: `git clone https://github.com/aws-samples/aws-cdk-golang-serverless-cicd-github-actions.git`	AWS DevOps
Vai al percorso Dockerfile.	Per accedere al percorso Dockerfile, esegui il seguente comando: `cd lambda`	AWS DevOps
Autentica Docker con Amazon ECR.	Amazon ECR richiede un accesso sicuro ai tuoi repository di container privati.	AWS DevOps

Operazione	Description	Competenze richieste
	Effettuando l'accesso in questo modo, consenti a Docker sul tuo computer o CI/CD ambiente locale di interagire con Amazon ECR in modo sicuro. Per autenticare Docker con Amazon ECR, esegui il seguente comando: `aws ecr get-login -password --region <AWS_REGION> | docker login --username AWS --password-stdin <AWS_ACCOUNT_ID>.dkr.ecr.<AWS_REGION>.amazonaws.com` Modifica i segnaposti e inserisci le tue informazioni `AWS_REGION` e `AWS_Account_ID`	
Creazione dell'immagine Docker.	Per creare l'immagine Docker, esegui il seguente comando: `docker build --platform linux/arm64 -t sample-app .`	AWS DevOps

Operazione	Description	Competenze richieste

Etichetta e invia l'immagine Docker.	Per taggare e inviare l'immagine Docker al repository Amazon ECR, esegui i seguenti comandi:
	`docker tag sample-app:latest <AWS_ACCOUNT_ID>.dkr.ecr.<AWS_REGION>.amazonaws.com/<ECR_REPOSITORY>:<DOCKER_TAG>`
	`docker push <AWS_ACCOUNT_ID>.dkr.ecr.<AWS_REGION>.amazonaws.com/<ECR_REPOSITORY>:<DOCKER_TAG>`
	Modifica i segnaposti e inserisci le tue informazioni. `AWS_Account_ID` `AWS_REGION` `ECR_REPOSITORY` `DOCKER_TAG`
		AWS DevOps

Distribuisci l'app AWS CDK

Operazione	Description	Competenze richieste
Sintetizza lo stack CDK con variabili specifiche dell'ambiente.	Per generare il CloudFormation modello per l'infrastruttura come definito nel codice CDK, esegui il comando seguente:	AWS DevOps

Operazione	Description	Competenze richieste
	`ENV=<environment> IMAGETAG=<image_tag> ECR_ARN=<ecr_repo_arn> cdk synth` Modifica i seguenti segnaposto con le tue informazioni: • `environment` — Sostituiscilo con un nome di ambiente specifico come `dev`, `staging` o. `prod` • `image_tag` — Sostituiscilo con un tag specifico un'immagine Docker come `v1.0.0` o `latest`. • `ecr_repo_arn` — Sostituisci con l'Amazon Resource Name (ARN) di un repository Amazon ECR.	

Operazione	Description	Competenze richieste
Implementa lo stack CDK.	Per distribuire lo stack CDK sul tuo Account AWS, esegui il comando seguente. Il *`--require-approval never`* flag indica che il CDK approverà ed eseguirà automaticamente tutte le modifiche. Ciò include le modifiche che il CDK normalmente contrasse gnerebbe come richiedenti una revisione manuale (come le modifiche alle politiche IAM o la rimozione di risorse). Assicurati che il codice CDK e la CI/CD pipeline siano ben testati e sicuri prima di utilizzar e il `--require-approval never` flag negli ambienti di produzione. `ENV=<environment> IMAGETAG=<image_ta g> ECR_ARN=<ecr_repo_ arn> cdk deploy -- require-approval never`	AWS DevOps

Automatizza CI/CD utilizzando i flussi di lavoro GitHub Actions

Operazione	Description	Competenze richieste
Crea un feature branch e aggiungi le tue modifiche.	Utilizzate il repository clonato che avete creato in precedenz a, create un feature branch,	AWS DevOps

Operazione	Description	Competenze richieste

	quindi aggiungete le modifiche al codice dell'applicazione. Utilizza il seguente comando:
	
	`git checkout -b <feature_branch>`
	`git add .`
	`git commit -m "add your changes"`
	`git push origin <feature_branch>`
	
	Di seguito sono riportati alcuni esempi di modifiche:
	
	• Modifiche alla logica della funzione Lambda
	• Aggiungere nuove caratteri stiche o funzionalità al codice Lambda
	• Correzione di bug o ottimizzazione del codice esistente all'interno della funzione Lambda
	
	GitHub Le azioni utilizzeranno i flussi di lavoro riutilizzabili e attiveranno le pipeline. CI/CD	

Unisci le modifiche.	Crea una pull request e unisci le modifiche alla pagina principale.	AWS DevOps

risoluzione dei problemi

Problema	Soluzione
`AccessDenied` errori durante la distribuzione di risorse su Account AWS, ad esempio,`AccessDenied: User not authorized to perform: "sts:AssumeRole"` .	Per risolvere questo problema, procedi come segue per verificare le autorizzazioni tra più account: • Assicurati che siano disponibili i ruoli e le policy IAM necessari per le distribuzioni tra account. • Verifica che le autorizzazioni dei `assume` ruoli siano configurate correttamente.
Problemi di compatibilità dovuti a mancate corrispondenze tra le versioni, ad esempio un `undefined: awscdkStack` errore con una versione CDK obsoleta.	Per risolvere questo problema, procedi come segue per verificare che stai utilizzando le versioni richieste di and Go: AWS CDK • Assicurati di utilizzare versioni compatibili di AWS CDK and Go. • Verifica la presenza di eventuali problemi noti o modifiche sostanziali nelle versioni recenti.
Guasti della pipeline CI/CD, ad esempio, a `Error: No such file or directory` causa di una configurazione YAML errata o di filiali protette. `Permission denied`	Per risolvere i problemi relativi alla configurazione GitHub Actions, verificate che i flussi di lavoro riutilizzabili siano referenziati e configurati correttamente.

Risorse correlate

Risorse AWS

- AWS Best practice per sicurezza, identità e conformità.
- AWS CDK Workshop
- AWS Libreria del Cloud Development Kit
- Creare una funzione Lambda utilizzando un'immagine del contenitore

- Identity and Access Management per Amazon Elastic Container Registry
- Lavorare con il CDK AWS in Go

Altre risorse

- Configurazione di OpenID Connect in Amazon Web GitHub Services (documentazione)
- Documentazione Golang
- Quickstart for GitHub Actions (documentazione) GitHub
- Riutilizzo dei flussi di lavoro (documentazione) GitHub

Esegui il provisioning di AWS Service Catalog prodotti basati su AWS CloudFormation modelli utilizzando GitHub Actions

Ashish Bhatt e Ruchika Modi, Amazon Web Services

Riepilogo

Questo modello offre alle organizzazioni un approccio semplificato che utilizza AWS Service Catalogprodotti e portafogli per fornire prodotti e portafogli standardizzati e conformi a tutti i team. Servizi AWS AWS CloudFormationaiuta a combinare componenti essenziali nei prodotti e nei portafogli di Service Catalog per il provisioning dell'infrastruttura di rete di base. Cloud AWSQuesto modello promuove anche DevOps pratiche che integrano l'infrastruttura come codice (IaC) nei flussi di lavoro di sviluppo automatizzati utilizzando Actions. GitHub

AWS Service Catalog consente alle organizzazioni di creare e gestire servizi IT approvati su AWS, offrendo vantaggi quali standardizzazione, controllo centralizzato, fornitura self-service e gestione dei costi. Automatizzando l'implementazione dei portafogli e dei prodotti Service Catalog tramite GitHub Actions, le aziende possono fare quanto segue:

- Ottieni implementazioni coerenti e ripetibili.
- Usa il controllo della versione per IAc.
- Integra la gestione delle risorse cloud con i flussi di lavoro di sviluppo esistenti.

Questa combinazione semplifica le operazioni cloud, rafforza la conformità e accelera la fornitura di servizi approvati, riducendo al contempo gli errori manuali e migliorando l'efficienza complessiva.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Accesso al GitHub repository
- Conoscenza di base di e AWS CloudFormation AWS Service Catalog
- Un bucket Amazon Simple Storage Service (Amazon S3) per ospitare modelli CloudFormation
- Un ruolo AWS Identity and Access Management (IAM) denominato utilizzato per `github-actions` la connettività tra e GitHub AWS

Limitazioni

- Il codice riutilizzabile di questo pattern è stato testato solo con GitHub Actions.
- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Versioni del prodotto

La soluzione di questo pattern è stata creata utilizzando le seguenti azioni del GitHub Marketplace e le rispettive versioni:

- `actions/checkout@v4`
- `aws-actions/configure-aws-credentials@v2`
- `aws-actions/aws-cloudformation-github-deploy@v1.2.0`

Architecture

Il diagramma seguente mostra l'architettura di questa soluzione.

1. Gli amministratori o gli ingegneri della piattaforma inviano CloudFormation modelli standardizzati a un GitHub repository, dove vengono mantenuti. Il GitHub repository contiene anche flussi di lavoro che automatizzano il provisioning dell'utilizzo di Actions. AWS Service Catalog GitHub
2. GitHub Actions attiva un flusso di lavoro che si connette all' Cloud AWS utilizzo di un provider OpenID Connect (OIDC) per la fornitura di Service Catalog.
3. Service Catalog contiene il portafoglio e i prodotti che gli sviluppatori possono utilizzare direttamente per fornire AWS risorse standardizzate. Questo modello raggruppa AWS risorse come cloud privati virtuali (VPCs), sottoreti, gateway NAT e Internet e tabelle di routing.
4. Dopo che lo sviluppatore ha creato un prodotto Service Catalog, Service Catalog lo converte in risorse preconfigurate e AWS standardizzate. Di conseguenza, gli sviluppatori risparmiano tempo perché non devono fornire risorse individuali e configurarle manualmente.

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita in e in tutto il loro ciclo di vita. Account AWS Regioni AWSÈ un servizio Infrastructure as Code (IaC) che può essere facilmente utilizzato come uno dei tipi di prodotto con. AWS Service Catalog
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Service Catalogti aiuta a gestire centralmente il catalogo di servizi IT approvati. AWS Gli utenti finali possono distribuire rapidamente soltanto i servizi IT approvati di cui hanno bisogno, in accordo con i vincoli stabiliti dall'organizzazione.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri

- GitHub Actions è una piattaforma di integrazione e distribuzione continua (CI/CD) strettamente integrata con i repository. GitHub Puoi utilizzare GitHub Actions per automatizzare la pipeline di compilazione, test e distribuzione.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub service-catalog-with-github-actions. Il repository contiene i seguenti file di interesse:

- `github/workflows`:
  - `e2e-test.yaml`— Questo file chiama`workflow.yaml`, che è il flusso di lavoro riutilizzabile. Questo flusso di lavoro viene attivato non appena viene eseguito un commit and push su una filiale.
  - `workflow.yaml`— Questo file contiene il flusso di lavoro riutilizzabile per questa soluzione ed è configurato `workflow_call` come trigger. Essendo un flusso di lavoro riutilizzabile, `workflow.yaml` può essere richiamato da qualsiasi altro flusso di lavoro.
- `templates`:
  - `servicecatalog-portfolio.yaml`— Questo CloudFormation modello include risorse che forniscono il portafoglio Service Catalog e il prodotto Service Catalog. Il modello contiene un set di parametri che vengono utilizzati durante il provisioning del portafoglio e dei prodotti Service Catalog. Un parametro accetta l'URL di un file Amazon S3 in cui `vpc.yaml` viene caricato il

modello. Sebbene questo modello includa il `vpc.yaml` file per il provisioning AWS delle risorse, puoi anche utilizzare il parametro URL del file S3 per la configurazione.

- `vpc.yaml`— Questo CloudFormation modello contiene AWS risorse da aggiungere nel prodotto Service Catalog. AWS le risorse includono VPCs sottoreti, gateway Internet, gateway NAT e tabelle di routing. Il `vpc.yaml` modello è un esempio di come è possibile utilizzare qualsiasi CloudFormation modello con un modello di prodotto e portafoglio Service Catalog.

Best practice

- Consulta le migliori pratiche di sicurezza AWS Service Catalog nella AWS Service Catalog documentazione.
- Vedi Rafforzamento della sicurezza per GitHub le azioni nella GitHub documentazione.

Epiche

Configura una workstation locale

Operazione	Description	Competenze richieste
Configurare Git sulla workstation locale.	Per installare e configurare Git sulla tua workstation locale, usa le istruzioni Getting Started — Installing Git nella documentazione di Git.	Sviluppatore di app
Clona il repository del GitHub progetto.	Per clonare il repository del GitHub progetto, procedi come segue: 1. Apri il GitHub repository per questo pattern 2. Scegli Code per visualizzare le opzioni di clonazione e copia l'URL fornito nella scheda HTTPS.	DevOps ingegnere

Operazione	Description	Competenze richieste

	3. Crea una cartella per il tuo progetto sulla tua workstation.
	4. Apri un terminale e accedi a questa cartella
	5. Per clonare il GitHub repository, esegui il seguente comando utilizzando l'URL che hai copiato nel passaggio 2:
	`git clone https://github.com/aws-samples/service-catalog-with-github-actions.git`
	6. Una volta completata la clonazione, per passare al repository clonato nella cartella del progetto, esegui il comando seguente:
	`cd <folder-name>/service-catalog-with-github-actions`
	7. Apri il progetto in un ambiente di sviluppo integrato (IDE) di tua scelta.	

Configura il provider OIDC

Operazione	Description	Competenze richieste
Configurare un provider OIDC.	Crea un provider OpenID Connect (OIDC) che consenta ai flussi di lavoro GitHub Actions di accedere alle risorse AWS, senza dover archiviare le credenziali come segreti di lunga durata. AWS GitHub Per istruzioni, consulta Configurazione di OpenID Connect in Amazon Web Services GitHub nella documentazione. Dopo la configurazione di un provider OIDC, la policy di fiducia del ruolo IAM`github-actions` , menzionata in precedenza nei Prerequisiti, verrà aggiornata.	Amministratore AWS, AWS DevOps, AWS generale

Pipeline Trigger GitHub Actions per implementare il portafoglio e i prodotti Service Catalog

Operazione	Description	Competenze richieste
Aggiornare `e2e-test.yaml` .	Il `e2e-test.yaml` file attiva il flusso di lavoro riutilizzabile in. `workflow.yaml` Aggiorna e convalida i valori per i seguenti parametri di input in: `e2e-test.yaml`	DevOps ingegnere

Operazione	Description	Competenze richieste

	• `aws_account_id` — Specificare il valore corretto Account AWS.
	• `aws_region` — Specificare il valore corretto Regione AWS.
	• `s3BucketName` — Specificare il bucket Amazon S3 per contenere i modelli. CloudFormation
	• Il file di workflow richiede due ruoli IAM come input:
	  • `LaunchConstraintRole` - Il ruolo IAM che AWS Service Catalog assume quando un utente finale avvia, aggiorna o chiude un prodotto.
	  • `PrincipalArn` - L'Amazon Resource Name (ARN) del principale (utente, ruolo o gruppo IAM) che verrà associato al Service Catalog Portfolio. In caso `PrincipalType` affermativo`IAM`, il valore supportato è un Amazon Resource Name (ARN) IAM completamente definito. Se `PrincipalType` è`IAM_PATTERN`, il valore supportato è un ARN IAM senza	

Operazione	Description	Competenze richieste
	un `AccountID` nel seguente formato: `arn:partition:iam::::resource-type/resource-id`	

Convalida l'implementazione

Operazione	Description	Competenze richieste
Convalida le risorse del Service Catalog.	Per convalidare le risorse del Service Catalog, procedi come segue: 1. Accedere a Console di gestione AWS di Account AWS e verificare che Regione AWS sia corretto. 2. Accedi AWS Service Cataloge verifica che un portafoglio sia presente in Amministrazione, Portafogli. 3. Scegliete il portafoglio e convalidate le informazi oni nelle schede Prodotti, Vincoli e Accesso.	AWS DevOps

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Elimina lo CloudFormation stack.	Per eliminare lo CloudForm ation stack, procedi come segue:	DevOps ingegnere, amministr atore AWS

Operazione	Description	Competenze richieste
	1. Apri la AWS CloudFormation console in https://console.aws.amazon.com/cloudformazione. 2. Nella barra di navigazione nella parte superiore dello schermo, scegli Regione AWS dove si trova lo stack. 3. Nell pagina Stack, seleziona lo stack che desideri eliminare. Lo stack deve essere attualmente in esecuzione. 4. Nel riquadro dei dettagli dello stack, scegliere Delete (Elimina). 5. Selezionare Delete stack (Elimina stack) quando richiesto. Per ulteriori informazioni, consulta Eliminare uno stack dalla CloudFormation console nella documentazione CloudFormation	

Risoluzione dei problemi

Problema	Soluzione
`e2e-test`	Per assicurarti di avere abilitato le impostazioni corrette del repository, procedi come segue:

Problema	Soluzione

`Can't find 'action.yml', 'action.yaml' or 'Dockerfile' under '*/home/runner/work/service-catalog-with-github-actions/service-catalog-with-github-actions`
 	
`Did you forget to run actions/checkout before running your local action?`
	1. Vai al repository Github, scheda Impostazioni.
	2. Scegli Azioni, Generali dal menu a sinistra
	3. Vai alla sezione Accesso e seleziona l'opzione Accessibile dai repository nell'organizzazione 'XXX'.

Risorse correlate

AWS documentazione

- Panoramica del Service Catalog

Altre risorse

- Informazioni sugli eventi che attivano i flussi di lavoro (GitHub documentazione)
- Riutilizzo dei flussi di lavoro (documentazione) GitHub

Informazioni aggiuntive

Per vedere gli screenshot relativi a Epics, vai alla cartella Immagini nel repository di questo pattern. GitHub Sono disponibili le schermate seguenti:

- AWS Service Catalog portfolio, sezione Amministrazione
- AWS Service Catalog prodotto, sezione Amministrazione
- AWS Service Catalog prodotto, User/Provisioning sezione

Fornisci ruoli IAM con privilegi minimi implementando una soluzione di distribuzione automatica di ruoli

Benjamin Morris, Nima Fotouhi, Aman Kaur Gandhi e Ciad Moon, Amazon Web Services

Riepilogo

Le autorizzazioni di ruolo sovradimensionate AWS Identity and Access Management (IAM) per le pipeline possono comportare rischi inutili per un'organizzazione. Gli sviluppatori a volte concedono ampie autorizzazioni durante lo sviluppo, ma trascurano di limitarle dopo la risoluzione dei problemi relativi al codice. Ciò causa un problema in cui sono presenti ruoli potenti senza necessità aziendali e potrebbe non essere mai stato esaminato da un tecnico della sicurezza.

Questo modello offre una soluzione a questo problema: il role vending machine (RVM). Utilizzando un modello di implementazione sicuro e centralizzato, RVM dimostra come fornire ruoli IAM con privilegi minimi per le pipeline dei singoli repository con il minimo sforzo da parte degli sviluppatori. GitHub Poiché RVM è una soluzione centrale, è possibile configurare i team di sicurezza come revisori necessari per approvare le modifiche. Questo approccio consente alla sicurezza di rifiutare le richieste di ruolo della pipeline sovraautorizzate.

La RVM accetta il codice Terraform come input e genera ruoli IAM pronti per la pipeline come output. Gli input richiesti sono l' Account AWS ID, il nome del repository e la politica delle autorizzazioni. GitHub La RVM utilizza questi input per creare la politica di fiducia e la politica di autorizzazione del ruolo. La policy di fiducia risultante consente al GitHub repository specificato di assumere il ruolo e utilizzarlo per le operazioni di pipeline.

La RVM utilizza un ruolo IAM (configurato durante il bootstrap). Questo ruolo dispone delle autorizzazioni necessarie per assumere un account role-provisioning-role in ogni account dell'organizzazione. Il ruolo è configurato tramite AWS Control Tower Account Factory for Terraform (AFT) o AWS CloudFormation StackSets. role-provisioning-rolesSono i ruoli che effettivamente creano i ruoli della pipeline per gli sviluppatori.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS.
- Un' GitHub organizzazione utilizzata per implementare l'infrastruttura come codice (IaC) tramite GitHub Actions. (non GitHub Enterprise/Premium/Ultimate sono obbligatori).

- Un AWS ambiente con più account. Non è necessario che questo ambiente ne faccia parte AWS Organizations.
- Un meccanismo per implementare un ruolo IAM in tutti Account AWS (ad esempio, AFT o CloudFormation StackSets).
- Terraform versione 1.3 o successiva installata e configurata.
- Terraform AWS Provider versione 4 o successiva installata e configurata.

Limitazioni

- Il codice di questo pattern è specifico di GitHub Actions e Terraform. Tuttavia, i concetti generali del pattern possono essere riutilizzati in altri framework di integrazione e distribuzione continua (CI/CD).
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità regionale, vedi AWS Servizi per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Architecture

Il diagramma seguente illustra il flusso di lavoro per questo modello.

Il flusso di lavoro per l'utilizzo tipico del distributore automatico di ruoli prevede i seguenti passaggi:

1. Uno sviluppatore invia il codice che contiene il codice Terraform per un ruolo IAM appena richiesto al repository RVM. GitHub Questa azione attiva la pipeline RVM Actions. GitHub
2. La pipeline utilizza una policy di trust OpenID Connect (OIDC) per assumere il ruolo di assunzione del ruolo RVM.
3. Durante l'esecuzione, la pipeline RVM assume il ruolo di workflow RVM nell'account in cui fornisce il nuovo ruolo IAM dello sviluppatore. (Il ruolo del workflow RVM è stato assegnato utilizzando AFT o.) CloudFormation StackSets
4. RVM crea il ruolo IAM dello sviluppatore con autorizzazioni e fiducia appropriate, in modo che il ruolo possa essere assunto da altre pipeline di applicazioni.
5. Gli sviluppatori di app possono configurare le proprie pipeline di app per assumere questo ruolo fornito da RVM.

Il ruolo creato include le autorizzazioni richieste dallo sviluppatore e una policy. `ReadOnlyAccess` Il ruolo è assunto solo dalle pipeline eseguite sul `main` ramo del repository specificato dallo sviluppatore. Questo approccio aiuta a garantire che per utilizzare il ruolo possano essere necessarie la protezione e le revisioni delle filiali.

Automazione e scalabilità

Le autorizzazioni con privilegi minimi richiedono attenzione ai dettagli per ogni ruolo assegnato. Questo modello riduce la complessità richiesta per creare questi ruoli, consentendo agli sviluppatori di creare i ruoli di cui hanno bisogno senza molto apprendimento o impegno aggiuntivi.

Tools (Strumenti)

Servizi AWS

- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Organizationsè un servizio di gestione degli account che ti aiuta a consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente.

Altri strumenti

- Git è un sistema di controllo delle versioni distribuito e open source. Include la possibilità di creare un account aziendale.
- GitHub Actions è una piattaforma di integrazione e distribuzione continua (CI/CD) strettamente integrata con i repository. GitHub Puoi utilizzare GitHub Actions per automatizzare la pipeline di compilazione, test e distribuzione.
- Terraform è uno strumento di infrastruttura come codice (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.

Deposito di codici

Il codice per questo pattern è disponibile nel GitHub role-vending-machinerepository.

Best practice

- Rendi facile la strada giusta e difficile quella sbagliata: rendi facile fare la cosa giusta. Se gli sviluppatori hanno difficoltà con il processo di provisioning di RVM, potrebbero tentare di creare

ruoli con altri mezzi, il che mina la natura centrale di RVM. Assicurati che il tuo team di sicurezza fornisca indicazioni chiare su come utilizzare RVM in modo sicuro ed efficace.

Dovresti anche impedire agli sviluppatori di fare la cosa sbagliata. Utilizza le politiche di controllo dei servizi (SCPs) o i limiti di autorizzazione per limitare i ruoli che possono creare altri ruoli. Questo approccio può aiutare a limitare la creazione di ruoli solo a RVM e ad altre fonti attendibili.

- Fornisci buoni esempi: inevitabilmente, alcuni sviluppatori adatteranno i ruoli esistenti nell'archivio RVM come modelli informali per la concessione delle autorizzazioni ai loro nuovi ruoli. Se disponete di esempi con autorizzazioni minime da cui possono copiare, ciò può ridurre il rischio che gli sviluppatori richiedano autorizzazioni ampie e con un elevato numero di jolly card. Se iniziate con ruoli altamente autorizzati e con molti jolly, il problema può moltiplicarsi col passare del tempo.
- Usa convenzioni e condizioni di denominazione: anche se uno sviluppatore non conosce tutti i nomi delle risorse che la sua applicazione creerà, dovrebbe comunque limitare le autorizzazioni dei ruoli utilizzando una convenzione di denominazione. Ad esempio, se stanno creando bucket Amazon S3, il valore della loro chiave di risorsa potrebbe essere `arn:aws:s3:::myorg-myapp-dev-*` tale che il loro ruolo non disponga di autorizzazioni oltre ai bucket che corrispondono a quel nome. L'applicazione della convenzione di denominazione tramite una policy IAM ha l'ulteriore vantaggio di migliorare la conformità alla convenzione di denominazione. Questo miglioramento si verifica perché non sarà consentita la creazione di risorse non corrispondenti.
- Richiedi revisioni delle pull request (PR): il valore della soluzione RVM è che crea una posizione centrale in cui è possibile esaminare i nuovi ruoli della pipeline. Tuttavia, questo design è utile solo se esistono barriere che aiutano a garantire che codice sicuro e di alta qualità venga inviato alla RVM. Proteggi le filiali utilizzate per distribuire il codice (ad esempio`main`) dai push diretti e richiedi l'approvazione di tutte le richieste di unione destinate ad esse.
- Configura ruoli di sola lettura: per impostazione predefinita, RVM fornisce una versione di ogni ruolo richiesto. `readonly` Questo ruolo può essere utilizzato in CI/CD pipeline che non scrivono dati, come un flusso di lavoro di pipeline. `terraform plan` Questo approccio aiuta a prevenire modifiche indesiderate se un flusso di lavoro di sola lettura non funziona correttamente.

  Per impostazione predefinita, la `ReadOnlyAccess` policy AWS gestita è associata sia ai ruoli di sola lettura che ai ruoli di lettura-scrittura. Questa politica riduce la necessità di iterazioni per determinare le autorizzazioni richieste, ma potrebbe essere eccessivamente permissiva per alcune organizzazioni. Se lo desideri, puoi rimuovere la politica dal codice Terraform.
- Concedi autorizzazioni minime: segui il principio del privilegio minimo e concedi le autorizzazioni minime richieste per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.

Epiche

Prepara l'ambiente

Operazione	Description	Competenze richieste
Copia il repository di esempio nella tua GitHub organizza zione.	Clona il repository di questo pattern o trasferiscilo alla tua GitHub organizzazione in modo da poterlo adattare alle tue esigenze. • Se scegli di clonare questo repository, puoi usare il seguente comando: `git clone https://github.com/aws-samples/role-vending-machine` • Se scegli di eseguire il fork del repository alla tua GitHub organizzazione, puoi usare il seguente comando: `gh repo fork aws-samples/role-vending-machine --org YOUR_ORGANIZATION_NAME`	DevOps ingegnere
Determina il Account AWS per la RVM.	Determina quale implement azione dell'infrastruttura utilizzare Account AWS per la RVM. Non utilizzare l'account di gestione o root.	Architetto del cloud

Operazione	Description	Competenze richieste
(Facoltativo) Consenti la creazione PRs delle pipeline dell'organizzazione.	**Note** Questo passaggio è necessario solo se desideri consentire la creazione PRs del `generate_providers_and_account_vars` flusso di lavoro. Per consentire la creazione delle pipeline della tua organizzazione PRs, segui i seguenti passaggi: 1. Passa a `https://github.com/organizations/YOUR_ORG/settings/actions` . 2. Seleziona Consenti GitHub azioni per creare e approvare le richieste pull. Per ulteriori informazioni, consulta la sezione Gestione GitHub delle impostazioni delle azioni per un repository nella GitHub documentazione.	DevOps ingegnere

Operazione	Description	Competenze richieste
Concedi autorizzazioni di sola lettura all'account RVM.	Crea una politica di delega nel tuo account di gestione che conceda al tuo account RVM autorizzazioni di sola lettura. Ciò consente ai GitHub flussi di lavoro RVM di generare dinamicamente un elenco degli account dell'organizzazione durante l'esecuzione dello script. AWS `generate_providers_and_account_vars.py` Utilizzate il codice seguente e sostituitelo `<YOUR RVM Account ID>` con l' Account AWS ID selezionato nel passaggio 2: `{ "Version": "2012-10-17", "Statement": [ { "Sid": "Statement", "Effect": "Allow", "Principal": { "AWS": "arn:aws:iam::<YOUR RVM Account ID>:root" }, "Action": [ "organizations:ListAccounts",`	Amministratore cloud

Operazione	Description	Competenze richieste

	<pre>"organizations:DescribeOrganization",
	"organizations:DescribeOrganizationalUnit",
	"organizations:ListRoots",
	"organizations:ListAWSServiceAccessForOrganization",
	"organizations:ListDelegatedAdministrators"
	],
	"Resource": "*"
	}
	]
	}</pre>	

Operazione	Description	Competenze richieste
Aggiorna i valori predefiniti dal repository di esempio.	Per configurare la RVM in modo che funzioni nel tuo ambiente specifico Regione AWS, procedi come segue: 1. Aggiorna `scripts/generate_providers_and_account_vars.py` e aggiorna altri file (come la `bootstrap` cartella) con la regione appropriata in cui operi. 2. Aggiorna il `.github/workflows/.env` file con l' Account AWS ID, la regione e tutte le altre variabili che desideri specificare.	DevOps ingegnere

Inizializza l'infrastruttura

Operazione	Description	Competenze richieste
Avvia il repository RVM.	Questo passaggio è necessario per creare i ruoli OIDC Trust e IAM utilizzati dalla pipeline RVM stessa, in modo che possa iniziare a operare e vendere altri ruoli. Nel contesto del tuo account RVM, esegui manualmente un comando dalla directory `. terraform apply`	DevOps ingegnere

Operazione	Description	Competenze richieste
	`scripts/bootstrap` Fornite tutti i valori richiesti in base alla documentazione delle variabili.	

Configura le operazioni

Operazione	Description	Competenze richieste
Distribuisci i `github-workflow-rvm-readonly` ruoli `github-workflow-rvm` and a tutti gli account.	Scegliete un metodo di distribuzione in linea con le pratiche della vostra organizzazione, come AFT o. StackSets Utilizza questo metodo per distribuire i due ruoli IAM nel `scripts/assumed_role/main.tf` file (nomi predefiniti `github-workflow-rvm` egithub-workflow-rvm-readonly) su ciascun account in cui desideri che RVM sia in grado di creare ruoli nella pipeline. Questi ruoli IAM dispongono di politiche di fiducia che consentono al ruolo di assunzione del ruolo dell'account RVM (o equivalente) di assumerlo. `readonly` I ruoli dispongono inoltre di politiche di autorizzazione IAM che consentono loro di leggere e scrivere (a meno che non si	Amministratore AWS

Operazione	Description	Competenze richieste
	utilizzi il `readonly` ruolo) ruoli corrispondenti. `github-workflow-role-*`	
Esegui il `generate_providers_and_account_vars` flusso di lavoro.	Per configurare la RVM in modo che sia pronta per la creazione di ruoli nella pipeline, procedi come segue: 1. Esegui il `generate_providers_and_account_vars` flusso di lavoro utilizzando workflow_dispatch. 2. Unisci il PR creato dal flusso di lavoro. Una volta completato il flusso di lavoro, la RVM è pronta per: • Accetta richieste per nuovi ruoli nella pipeline. • Crea ruoli di sola lettura o di lettura-scrittura come richiesto. • Distribuisci i ruoli nel modo specificato. Account AWS	DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
Ho creato un ruolo utilizzando RVM, ma GitHub non sono in grado di assumerlo.	Verifica che il nome del GitHub repository corrisponda al nome fornito al modulo.

Problema	Soluzione
	`github_workflow_roles` I ruoli sono definiti in modo che solo un repository possa assumerli. Analogamente, verificate che il ramo utilizzato nella GitHub pipeline corrisponda al nome del ramo fornito al modulo. `github_workflow_roles` In genere, i ruoli creati da RVM con autorizzazioni di scrittura possono essere utilizzati solo dai flussi di lavoro limitati al `main` ramo (ovvero, distribuzioni provenienti da). `main`
Il mio ruolo di sola lettura non riesce a eseguire la pipeline perché non dispone delle autorizzazioni per leggere una risorsa specifica.	Sebbene la `ReadOnlyAccess` politica fornisca ampie autorizzazioni di sola lettura, la politica non include alcune azioni di lettura (ad esempio, determinate azioni). AWS Security Hub CSPM È possibile aggiungere autorizzazioni per azioni specifiche utilizzando il `inline_policy_readonly` parametro del modulo. `github-workflow-roles`

Risorse correlate

- Le migliori pratiche per l'utilizzo AWS CloudFormation StackSets
- Organizzazione dell' AWS ambiente utilizzando più account
- Panoramica di AWS Control Tower Account Factory for Terraform (AFT)
- Le migliori pratiche politiche

Informazioni aggiuntive

Utilizzo GitHub degli ambienti

GitHub gli ambienti sono un approccio alternativo alle restrizioni basate sulle filiali per l'accesso ai ruoli. Se preferisci utilizzare un GitHub ambiente, di seguito è riportato un esempio della sintassi per una condizione aggiuntiva nella policy di fiducia IAM. Questa sintassi specifica che il ruolo può essere utilizzato solo quando l' GitHub azione è in esecuzione nell'ambiente. `Production`

```
"StringLike": {
    "token.actions.githubusercontent.com:sub": "repo:octo-org/octo-
repo:environment:Production"
}
```

La sintassi di esempio utilizza i seguenti valori segnaposto:

- `octo-org`è il nome dell'organizzazione. GitHub
- `octo-repo`è il nome del repository.
- `Production`è il nome specifico GitHub dell'ambiente.

Pubblica i CloudWatch parametri di Amazon in un file CSV

Abdullahi Olaoye, Amazon Web Services

Riepilogo

Questo modello utilizza uno script Python per recuperare le metriche di CloudWatch Amazon e convertire le informazioni sulle metriche in un file con valori separati da virgole (CSV) per una migliore leggibilità. Lo script accetta il servizio AWS le cui metriche devono essere recuperate come argomento obbligatorio. Puoi specificare la regione AWS e il profilo di credenziali AWS come argomenti opzionali. Se non specifichi questi argomenti, lo script utilizza la regione e il profilo predefiniti configurati per la workstation in cui viene eseguito lo script. Dopo l'esecuzione, lo script genera e archivia un file CSV nella stessa directory.

Vedi la sezione Allegati per lo script e i file associati forniti con questo modello.

Prerequisiti e limitazioni

Prerequisiti

- Python 3.x
- Interfaccia a riga di comando di AWS (CLI AWS)

Limitazioni

Lo script attualmente supporta i seguenti servizi AWS:

- AWS Lambda
- Amazon Elastic Compute Cloud (Amazon EC2)
  - Per impostazione predefinita, lo script non raccoglie i parametri di volume di Amazon Elastic Block Store (Amazon EBS). Per raccogliere i parametri di Amazon EBS, devi modificare il file allegato`metrics.yaml`.
- Amazon Relational Database Service (Amazon RDS)
  - Tuttavia, lo script non supporta Amazon Aurora.
- Application Load Balancer
- Network Load Balancer
- Gateway Amazon API

Tools (Strumenti)

- Amazon CloudWatch è un servizio di monitoraggio creato per DevOps ingegneri, sviluppatori, ingegneri dell'affidabilità del sito (SREs) e responsabili IT. CloudWatch fornisce dati e approfondimenti utilizzabili per aiutarti a monitorare le tue applicazioni, rispondere ai cambiamenti delle prestazioni a livello di sistema, ottimizzare l'utilizzo delle risorse e ottenere una visione unificata dello stato operativo. CloudWatch raccoglie dati operativi e di monitoraggio sotto forma di log, metriche ed eventi e fornisce una visione unificata delle risorse, delle applicazioni e dei servizi AWS eseguiti su server AWS e locali.

Epiche

Installa e configura i prerequisiti

Operazione	Description	Competenze richieste
Installa i prerequisiti.	Esegui il comando seguente: `$ pip3 install -r requirements.txt`	Developer
Configurare .	Esegui il comando seguente: `$ aws configure`	Developer

Configurare lo script Python

Operazione	Description	Competenze richieste
Apri lo script.	Per modificare la configura zione predefinita dello script, apri`metrics.yaml` .	Developer
Imposta il periodo per lo script.	Questo è il periodo di tempo da recuperare. Il periodo predefinito è di 5 minuti (300	Developer

Operazione	Description	Competenze richieste
	secondi). Puoi modificare il periodo di tempo, ma tieni presente le seguenti limitazioni: • Se il valore delle ore specificato è compreso tra 3 ore e 15 giorni fa, utilizza un multiplo di 60 secondi (1 minuto) per il periodo. • Se il valore delle ore specificato è compreso tra 15 ore e 63 giorni fa, utilizza un multiplo di 300 secondi (5 minuti) per il periodo. • Se il valore delle ore specificato è superiore a 63 giorni fa, utilizza un multiplo di 3.600 secondi (1 ora) per il periodo. In caso contrario, l'operazione API non restituirà alcun punto dati.	
Imposta le ore dello script.	Questo valore specifica quante ore di metriche vuoi recuperare. Il valore predefinito è 1 ora. Per recuperare più giorni di metriche, fornisci il valore in ore. Ad esempio, per 2 giorni, specifica 48.	Developer

Operazione	Description	Competenze richieste
Modifica i valori delle statistic he per lo script.	(Facoltativo) Il valore delle statistiche globali è utilizzat o per recuperare metriche a cui non è assegnato un valore statistico specifico. `Average` Lo script supporta i valori statistici `MaximumSampleCou nt` , e. `Sum`	Developer

Esegui lo script Python

Operazione	Description	Competenze richieste

Eseguire lo script.	Utilizza il seguente comando:
	`$ python3 cwreport.py <service>`
	Per visualizzare un elenco dei valori del servizio e dei `profile` parametri opzionali `region` , esegui il comando seguente:
	`$ python3 cwreport.py -h`
	Per ulteriori informazioni sui parametri opzionali, vedere la sezione Informazioni aggiuntiv e.
		Developer

Risorse correlate

- Configurazione dell'interfaccia a riga di comando di AWS
- Utilizzo dei CloudWatch parametri di Amazon
- CloudWatch Documentazione Amazon
- EC2 CloudWatch Parametri
- Metriche di AWS Lambda
- Metriche di Amazon RDS
- Metriche dell'Application Load Balancer
- Metriche di Network Load Balancer
- Metriche di Amazon API Gateway

Informazioni aggiuntive

Utilizzo degli script

```
$ python3 cwreport.py -h
```

Sintassi di esempio

```
python3 cwreport.py <service> <--region=Optional Region> <--profile=Optional credential
 profile>
```

Parametri

- service (richiesto) – Il servizio su cui si desidera eseguire lo script. Lo script attualmente supporta questi servizi: AWS Lambda, Amazon, EC2 Amazon RDS, Application Load Balancer, Network Load Balancer e API Gateway.
- region (opzionale) – La regione AWS da cui recuperare le metriche. La regione predefinita è. `ap-southeast-1`
- profile (opzionale) – Il profilo denominato della CLI AWS da utilizzare. Se questo parametro non è specificato, viene utilizzato il profilo di credenziali configurato di default.

Esempi

- Per utilizzare la regione `ap-southeast-1` e le credenziali configurate predefinite per recuperare i parametri Amazon EC2 : `$ python3 cwreport.py ec2`
- Per specificare una regione e recuperare le metriche dell'API Gateway: `$ python3 cwreport.py apigateway --region us-east-1`
- Per specificare un profilo AWS e recuperare i EC2 parametri Amazon: `$ python3 cwreport.py ec2 --profile testprofile`
- Per specificare sia la regione che il profilo per recuperare i EC2 parametri di Amazon: `$ python3 cwreport.py ec2 --region us-east-1 --profile testprofile`

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Rimuovi Amazon EC2 Entries Account AWS dall'altra AWS Managed Microsoft AD parte utilizzando l' AWS Lambda automazione

Dott. Rahul Sharad Gaikwad e Tamilselvan P, Amazon Web Services

Riepilogo

Active Directory (AD) è uno strumento di scripting di Microsoft che gestisce le informazioni sul dominio e le interazioni degli utenti con i servizi di rete. È ampiamente utilizzato dai fornitori di servizi gestiti (MSPs) per gestire le credenziali dei dipendenti e le autorizzazioni di accesso. Poiché gli aggressori di AD possono utilizzare account inattivi per cercare di hackerare un'organizzazione, è importante trovare account inattivi e disattivarli secondo un programma di manutenzione ordinaria. Con AWS Directory Service for Microsoft Active Directory, puoi eseguire Microsoft Active Directory come servizio gestito. Questo modello può aiutarti a configurare AWS Lambda l'automazione per trovare e rimuovere rapidamente gli account inattivi.

Se i seguenti scenari si applicano alla vostra organizzazione, questo modello può aiutarvi:

- Gestione AD centralizzata: se la tua organizzazione ne ha più di una Account AWS, ognuna con la propria implementazione AD, può essere difficile gestire gli account utente e le autorizzazioni di accesso in modo uniforme su tutti gli account. Con una soluzione di pulizia di AD per più account, puoi disabilitare o rimuovere gli account inattivi da tutte le istanze AD in modo centralizzato.
- Ristrutturazione o migrazione di AD: se la tua organizzazione prevede di ristrutturare o migrare la distribuzione di AD, una soluzione di pulizia di AD per più account può aiutarti a preparare l'ambiente. La soluzione può aiutarti a rimuovere gli account non necessari o inattivi, semplificare il processo di migrazione e ridurre potenziali conflitti o problemi.

Quando si utilizza questo modello, è possibile ottenere i seguenti vantaggi:

- Migliora le prestazioni di database e server e correggi le vulnerabilità di sicurezza dovute agli account inattivi.
- Se il server AD è ospitato nel cloud, la rimozione degli account inattivi può anche ridurre i costi di archiviazione migliorando al contempo le prestazioni. Le bollette mensili potrebbero diminuire perché sia i costi della larghezza di banda che le risorse di elaborazione possono diminuire.
- Tieni a bada i potenziali aggressori con un Active Directory pulito.

Prerequisiti e limitazioni

Prerequisiti

- Un account genitore attivo Account AWS e uno o più account per figli. In questo modello, un account principale è il luogo in cui viene creato Active Directory. Gli account secondari ospitano server Windows e vengono uniti tramite l'account principale Active Directory.
- Git installato e configurato su una workstation locale.
- Terraform installato e configurato su una workstation locale.
- AWS Managed Microsoft AD directory configurata nell'account principale e condivisa con tutti gli account secondari. Per maggiori dettagli, consulta Tutorial: Sharing your AWS Managed Microsoft AD directory for seamless EC2 domain-join nella Administration Guide.AWS Directory Service
- Una connessione peering di cloud privato virtuale (VPC) o una AWS Transit Gateway connessione disponibile tra il VPC di AWS Directory Service (account principale) e il VPC delle istanze Amazon Elastic Compute Cloud (Amazon EC2) (account secondari). Per ulteriori dettagli, consulta Configurare una connessione peering VPC tra il proprietario della directory e l'account utente della directory nella Guida all'AWS Directory Service amministrazione.
- Una macchina Windows configurata con lo `EC2WindowsUserdata` script su tutti gli account padre e figlio. Il file di script è disponibile nella radice del repository di codice di questo pattern.
- Un ruolo cross-account AWS Identity and Access Management (IAM) disponibile su ogni account secondario configurato con una policy di fiducia per consentire l'uso di una AWS Lambda funzione dell'account principale. Per ulteriori informazioni, consulta Invio e ricezione di eventi tra Amazon Account AWS EventBridge nella Amazon EventBridge User Guide.
- I seguenti valori segreti sono disponibili in AWS Systems Manager Parameter Store dell'account principale:
  - `domainJoinUser`— Nome utente del servizio di directory
  - `domainJoinPassword`— Password del servizio di elenco

  Per ulteriori informazioni sui segreti, vedere Create an Gestione dei segreti AWS secret nella Guida per l'Gestione dei segreti AWS utente.

Limitazioni

- La creazione di una risorsa in un account per bambini non è automatizzata con Terraform. È necessario creare manualmente le seguenti risorse utilizzando: Console di gestione AWS

  - EventBridge Regola di Amazon per inviare gli eventi di EC2 cessazione di Amazon all'account principale
  - Creazione EC2 di ruoli su più account Amazon nell'account per bambini con politica di fiducia
  - Peering VPC o connessione Transit Gateway
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- Terraform versione 1.1.9 o successiva
- Terraform AWS Provider versione 3.0 o successiva

Architecture

Il diagramma seguente mostra l'architettura di alto livello della soluzione.

Il diagramma dell'architettura illustra il seguente processo:

1. Negli account per bambini, la EventBridge regola raccoglie tutti gli eventi di EC2 cessazione di Amazon. La regola invia gli eventi a EventBridge cui è presente nell'account principale.
2. Dall'account principale, EventBridge raccoglie tutti gli eventi e contiene la regola per attivare la funzione `ADcleanup-Lambda` Lambda.
3. L'account principale riceve eventuali eventi di chiusura dall'account genitore o figlio e attiva la funzione Lambda.
4. La funzione Lambda effettua la chiamata ai gruppi di Amazon Auto EC2 Scaling utilizzando il modulo di avvio Python e ottiene l'ID casuale dell'istanza. L'ID dell'istanza viene utilizzato per eseguire i comandi di Systems Manager.
5. La funzione Lambda effettua un'altra chiamata ad Amazon EC2 utilizzando il modulo boto. La funzione Lambda ottiene gli indirizzi IP privati dei server Windows in esecuzione e li archivia in una variabile temporanea. Nei passaggi 5.1 e 5.2, le EC2 istanze di Windows in esecuzione vengono raccolte dagli account secondari.
6. La funzione Lambda effettua un'altra chiamata a Systems Manager per ottenere le informazioni sul computer a cui è connesso. AWS Directory Service

7. Un AWS Systems Manager documento aiuta a eseguire il PowerShell comando sui server Amazon EC2 Windows per ottenere gli indirizzi IP privati dei computer collegati ad AD. (Il documento Systems Manager utilizza l'ID dell'istanza ottenuto nel passaggio 4.)
8. Il nome utente e le password del dominio AD vengono archiviati nel AWS Systems Manager Parameter Store. AWS Lambda e Systems Manager effettuano una chiamata a Parameter Store e ottengono i valori di nome utente e password da utilizzare per connettersi ad AD.
9. Utilizzando il documento Systems Manager, lo PowerShell script viene eseguito sul server Amazon EC2 Windows utilizzando l'id dell'istanza ottenuto in precedenza nel passaggio 4.
10. Amazon EC2 si connette AWS Directory Service utilizzando PowerShell i comandi e rimuove i computer non in uso o inattivi.

Tools (Strumenti)

AWS servizi

- AWS Directory Serviceoffre diversi modi per utilizzare Microsoft Active Directory (AD) con altri Servizi AWS come Amazon Elastic Compute Cloud (Amazon EC2), Amazon Relational Database Service (Amazon RDS) per SQL Server e FSx Amazon per Windows File Server.
- AWS Directory Service for Microsoft Active Directoryconsente ai carichi di lavoro e alle risorse compatibili con le directory di AWS utilizzare Microsoft Active Directory in. Cloud AWS
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altro modo. Account AWS
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle. Con IAM, puoi specificare chi o cosa può accedere ai servizi e alle risorse AWS, gestire centralmente le autorizzazioni granulari e analizzare l'accesso per perfezionare le autorizzazioni in tutto il mondo. AWS
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.

- AWS Systems Managerti aiuta a gestire le applicazioni e l'infrastruttura in esecuzione in. Cloud AWS Semplifica la gestione delle applicazioni e delle risorse, riduce i tempi di rilevamento e risoluzione dei problemi operativi e aiuta a gestire le AWS risorse in modo sicuro su larga scala.
- AWS Systems Manager i documenti definiscono le azioni che Systems Manager esegue sulle istanze gestite. Systems Manager include più di 100 documenti preconfigurati che è possibile utilizzare specificando i parametri nel runtime (tempo di esecuzione).
- AWS Systems Manager Parameter Store è una funzionalità AWS Systems Manager e fornisce uno storage gerarchico sicuro per la gestione dei dati di configurazione e la gestione dei segreti.

Altri strumenti

- HashiCorp Terraform è uno strumento Infrastructure as Code (IaC) che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud.
- PowerShellè un programma di gestione dell'automazione e della configurazione di Microsoft che funziona su Windows, Linux e macOS.
- Python è un linguaggio di programmazione per computer generico.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub aws-lambda-ad-cleanup-terraform-samples.

Best practice

- Unisci automaticamente i domini.Quando avvii un'istanza Windows che deve far parte di un Directory Service dominio, unisciti al dominio durante il processo di creazione dell'istanza anziché aggiungere manualmente l'istanza in un secondo momento. Per aggiungere automaticamente un dominio, seleziona la directory corretta dall'elenco a discesa Domain join directory quando avvii una nuova istanza. Per maggiori dettagli, consulta Unire senza problemi un'istanza Amazon EC2 Windows alla tua AWS Managed Microsoft AD Active Directory nella Guida all'Directory Service amministrazione.
- Eliminare gli account non utilizzati.È comune trovare account in AD che non sono mai stati utilizzati. Come gli account disabilitati o inattivi che rimangono nel sistema, gli account trascurati e inutilizzati possono rallentare il sistema AD o rendere l'organizzazione vulnerabile alle violazioni dei dati.

- Automatizza le pulizie di Active Directory.Per contribuire a mitigare i rischi per la sicurezza e impedire che gli account obsoleti influiscano sulle prestazioni di AD, è necessario eseguire le pulizie di AD a intervalli regolari. Puoi eseguire la maggior parte delle attività di gestione e pulizia di AD scrivendo script. Le attività di esempio includono la rimozione di account disabilitati e inattivi, l'eliminazione di gruppi vuoti e inattivi e l'individuazione di account utente e password scaduti.

Poemi epici

Configura account per bambini

Operazione	Description	Competenze richieste
Crea un ruolo per più account nell'account per bambini.	Per creare un ruolo tra più account in un account figlio, procedi come segue: 1. Per ogni account figlio, crea un ruolo denominato `ec2crossaccountrole` utilizzando la politica gestita denominat `aAmazonEC2ReadOnlyAccess` . (Per ulteriori informazioni, consulta Creare un ruolo utilizzando policy di fiducia personalizzate nella documentazione IAM.) 2. Nella sezione Custom Trust Policy, aggiungi il seguente codice: `{ "Version": "2012-10-17", "Statement": [ {`	DevOps ingegnere

Operazione	Description	Competenze richieste
	"Effect": "Allow", "Principal": { "Service": "ec2.amazonaws.com" }, "Action": "sts:AssumeRole" }, { "Effect": "Allow", "Principal": { "AWS": "arn:aws:iam::${Parentaccountid}:role/ADcleanuprole" }, "Action": "sts:AssumeRole" }] }	

Operazione	Description	Competenze richieste
Crea una regola relativa agli eventi nell'account per bambini.	Per creare una EventBridge regola per ogni account per bambini, procedi come segue: 1. Accedi al bambino Account AWS, quindi apri la EventBridge console Amazon all'indirizzo https://console.aws.amazon.com/events/. 2. Nel pannello di navigazione, scegli Regole. 3. Scegli Crea regola. 4. Inserisci un nome e, facoltativamente, una descrizione per la regola. 5. Per Event bus, seleziona AWS default event bus. 6. Per Tipo di regola, scegli Regola con un pattern di eventi. 7. Scegli Next (Successivo). 8. Per Event pattern, incolla il seguente codice: `{ "source": ["aws.ec2"], "detail-type": ["EC2 Instance State-change Notification"], "detail": { "state": ["terminated"]`	DevOps ingegnere

Operazione	Description	Competenze richieste
	<pre> } }</pre> 9. Scegli Next (Successivo). 10. Per Target type, scegli Event bus in un account o in una regione diversi.Per Event bus come destinazione, inserisci il bus di eventi Amazon Resource Name (ARN) per l'account principale. 11. Per Ruolo di esecuzione, scegli Crea un nuovo ruolo per questa risorsa specifica. 12. Scegli Avanti per esaminare i dettagli della nuova regola, quindi scegli Crea. Per maggiori dettagli, consulta Creazione di regole che reagiscono agli eventi in Amazon EventBridge nella Amazon EventBridge User Guide.	

Operazione	Description	Competenze richieste
Crea un' EC2 istanza e unisciti ad AD.	Per creare un' EC2 istanza per Windows, procedi come segue: 1. Utilizzate lo `EC2WindowsUserdata` script disponibile nell'archivio di codice di questo pattern. 2. Nello script dei dati utente, rivedi il codice seguente per utilizzare `Directory service addresses` i valori dell'account principale: `set-DnsClientServerAddress -InterfaceIndex 6 -ServerAddresses $(Directory service addresses)`	DevOps ingegnere

Configura la workstation locale

Operazione	Description	Competenze richieste
Crea una cartella di progetto e aggiungi i file.	Per clonare il repository e creare una cartella di progetto, procedi come segue: 1. Apri l'archivio di questo pattern. GitHub 2. Scegli il pulsante Codice per visualizzare le opzioni	DevOps ingegnere

Operazione	Description	Competenze richieste

	per la clonazione nel menu a discesa Clone.
	3. Nella scheda HTTPS, copia l'URL fornito in Clone utilizzando l'URL web.
	4. Create una cartella sul computer e denominatela con il nome del progetto.
	5. Apri un terminale nel tuo computer locale e accedi a questa cartella.
	6. Per clonare il repository git, usa il seguente comando.
	`git clone <repository-URL>.git`
	7. Dopo che il repository è stato clonato, usa il seguente comando per andare alla directory clonata.
	`cd <directory name>/terraform-aws-lambda-ad-cleanup/multiple-account-cleanup`
	8. Nel repository clonato, apri questo progetto in un ambiente di sviluppo integrato (IDE) di tua scelta.	

Operazione	Description	Competenze richieste
Crea il `adcleanup.zip` file.	Per comprimere il `lambda_function.py` file, esegui il seguente comando: `zip -r adcleanup.zip lambda_function.py`	DevOps ingegnere

Fornisci l'architettura di destinazione utilizzando la configurazione Terraform

Operazione	Description	Competenze richieste
Fornisci valori per le variabili Terraform.	Per l'account secondario, fornisci i valori per le seguenti `arn` variabili come tipi di stringa nel `terraform.tfvars` file: • `lambda_env_cross_role_arn` • `child_account_cross_role_arn`	DevOps ingegnere
Inizializza la configurazione Terraform.	Per inizializzare la directory di lavoro che contiene i file Terraform, esegui il seguente comando: `terraform init`	DevOps ingegnere
Visualizza in anteprima le modifiche.	Puoi visualizzare in anteprima le modifiche che Terraform apporterà all'infrastruttura prima che l'infrastruttura venga implementata. Per verificare che Terraform apporti le	DevOps ingegnere

Operazione	Description	Competenze richieste
	modifiche necessarie, esegui il seguente comando: `terraform plan –– var-file=examples/ terraform.tfvars`	
Esegui le azioni proposte.	Per verificare che i risultati del `terraform plan` comando siano quelli previsti, effettuate le seguenti operazioni: 1. Esegui il comando riportato qui di seguito: `terraform apply` 2. Accedere a Console di gestione AWS e verificare che le risorse siano presenti.	DevOps ingegnere

Verifica della distribuzione

Operazione	Description	Competenze richieste
Esegui e testa la funzione Lambda.	Per verificare che la distribuzione sia avvenuta correttamente, procedi come segue: 1. Accedi a e Console di gestione AWS apri la console Lambda. Aprire la pagina Funzioni e selezionare il nome della funzione che inizia con ADcleanup-Lambda-*.	DevOps ingegnere

Operazione	Description	Competenze richieste
	2. Nella pagina di panoramica delle funzioni, scegliete Test nella scheda Codice nella sezione Codice sorgente. 3. Per salvare l'evento di test, fornite un nome per l'evento e scegliete Salva. Per testare l'evento, scegli nuovamente Test. I risultati dell'esecuzione mostrano l'output della funzione.	

Operazione	Description	Competenze richieste
Visualizza i risultati dell'esecuzione delle EventBridge regole dall'account principale.	Per visualizzare i risultati della EventBridge regola basata sugli eventi di EC2 cessazione di Amazon dall'account principale, procedi come segue: 1. Termina le EC2 istanze dall'account principale. 2. Apri la console Lambda dell'account principale. Aprire la pagina Funzioni e selezionare il nome della funzione che inizia con ADcleanup-Lambda-*. 3. Scegliete la scheda Monitor e scegliete Visualizza registri. CloudWatch Nella CloudWatch console, la pagina Log groups mostra i risultati della funzione Lambda.	DevOps ingegnere

Operazione	Description	Competenze richieste
Visualizza i risultati dell'esecuzione delle EventBridge regole dall'account secondario.	Per visualizzare i risultati della EventBridge regola basata sugli eventi di EC2 chiusura di Amazon dall'account secondario, procedi come segue: 1. Termina le EC2 istanze dall'account secondario. 2. Apri la console Lambda dell'account principale. Aprire la pagina Funzioni e selezionare il nome della funzione che inizia con ADcleanup-Lambda-*. 3. Scegliete la scheda Monitor e scegliete Visualizza registri. CloudWatch Nella CloudWatch console, la pagina Log groups mostra i risultati della funzione Lambda.	DevOps ingegnere

Pulisci l'infrastruttura dopo l'uso

Operazione	Description	Competenze richieste
Pulisci l'infrastruttura.	Per pulire l'infrastruttura che hai creato, usa il seguente comando: `terraform destroy`	DevOps ingegnere

Operazione	Description	Competenze richieste
	Per confermare il `destroy` comando, digitate`yes`.	
Verifica dopo la pulizia.	Verifica che le risorse siano state rimosse correttamente.	DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
Problema di connessione tra AWS Directory Service (account principale) e EC2 istanza Amazon (account figlio): non è possibile unire i computer dell'account secondario ad AD anche se è disponibile il peering VPC.	Aggiungi il routing in. VPCs Per istruzioni, consulta Configurare una connessione peering VPC tra il proprietario della directory e l'account utente della directory nella documentazione. AWS Directory Service

Risorse correlate

AWS documentazione

- Amazon EventBridge e AWS Identity and Access Management
- Configurazione delle autorizzazioni dell'istanza richieste per Systems Manager
- Gestione delle identità e degli accessi per Directory Service
- Policy IAM basate sull'identità per Lambda
- Aggiungi manualmente un'istanza Amazon EC2 Windows alla tua AWS Managed Microsoft AD Active Directory
- Rimuovi EC2 le voci Amazon nello stesso Account AWS modulo AWS Managed Microsoft AD utilizzando l' AWS Lambda automazione

Altre risorse

- AWS Provider (documentazione Terraform)
- Configurazione del backend (documentazione Terraform)

- Installa Terraform (documentazione Terraform)
- Modulo di avvio Python (repository Python Package Index)
- Download del file binario Terraform (documentazione Terraform)

Rimuovi EC2 le voci Amazon nello stesso Account AWS modulo AWS Managed Microsoft AD utilizzando l' AWS Lambda automazione

Dott. Rahul Sharad Gaikwad e Tamilselvan P, Amazon Web Services

Riepilogo

Active Directory (AD) è uno strumento di scripting di Microsoft che gestisce le informazioni sul dominio e le interazioni degli utenti con i servizi di rete. È ampiamente utilizzato dai fornitori di servizi gestiti (MSPs) per gestire le credenziali dei dipendenti e le autorizzazioni di accesso. Poiché gli aggressori di AD possono utilizzare account inattivi per cercare di hackerare un'organizzazione, è importante trovare account inattivi e disattivarli secondo un programma di manutenzione ordinaria. Con AWS Directory Service for Microsoft Active Directory, puoi eseguire Microsoft Active Directory come servizio gestito.

Questo modello può aiutarti a configurare AWS Lambda l'automazione per trovare e rimuovere rapidamente gli account inattivi. Quando si utilizza questo modello, è possibile ottenere i seguenti vantaggi:

- Migliora le prestazioni di database e server e correggi le vulnerabilità di sicurezza dovute agli account inattivi.
- Se il server AD è ospitato nel cloud, la rimozione degli account inattivi può anche ridurre i costi di archiviazione migliorando al contempo le prestazioni. Le bollette mensili potrebbero diminuire perché sia i costi della larghezza di banda che le risorse di elaborazione possono diminuire.
- Tieni a bada i potenziali aggressori con un Active Directory pulito.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Git installato e configurato su una workstation locale.
- Terraform installato e configurato su una workstation locale.
- Computer Windows con moduli Active Directory ()`ActiveDirectory`.
- Una directory in AWS Managed Microsoft AD e credenziali memorizzate in un AWS Systems Manager parametro in Parameter Store.

- AWS Identity and Access Management Ruolo (IAM) con autorizzazioni per l' Servizi AWS elenco in Strumenti. Per ulteriori informazioni su IAM, consulta Risorse correlate.

Limitazioni

- Questo modello non supporta la configurazione tra account.
- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- Terraform versione 1.1.9 o successiva
- Terraform AWS Provider versione 3.0 o successiva

Architecture

Il diagramma seguente mostra i componenti del flusso di lavoro e dell'architettura per questo modello.

Il diagramma mostra il flusso di lavoro seguente:

1. Amazon EventBridge attiva la AWS Lambda funzione in base a un'espressione cron. (Per questo modello, la pianificazione delle espressioni cron è una volta al giorno.)
2. Il ruolo e la policy IAM richiesti vengono creati e collegati AWS Lambda tramite Terraform.
3. La AWS Lambda funzione viene eseguita e chiama i gruppi di Auto Scaling di Amazon Elastic Compute Cloud (Amazon EC2) utilizzando il modulo boto Python. La funzione Lambda ottiene l'id casuale dell'istanza. L'id dell'istanza viene utilizzato per eseguire AWS Systems Manager i comandi.
4. AWS Lambda effettua un'altra chiamata ad Amazon EC2 utilizzando il modulo boto e ottiene gli indirizzi IP privati dei server Windows in esecuzione e li archivia in una variabile temporanea.
5. AWS Lambda effettua un'altra chiamata a Systems Manager per ottenere le informazioni sul computer a cui è connesso Directory Service.
6. Un AWS Systems Manager documento aiuta a eseguire lo PowerShell script sui server Amazon EC2 Windows per ottenere gli indirizzi IP privati dei computer collegati ad AD.

7. Il nome utente e le password del dominio AD sono archiviati nel AWS Systems Manager Parameter Store. AWS Lambda e Systems Manager effettuano una chiamata a Parameter Store e ottengono i valori di nome utente e password da utilizzare per connettere AD.
8. Utilizzando il documento Systems Manager, lo PowerShell script viene eseguito sul server Amazon EC2 Windows utilizzando l'id dell'istanza ottenuto in precedenza nel passaggio 3.
9. Amazon EC2 si connette Directory Service utilizzando PowerShell i comandi e rimuove i computer non in uso o inattivi.

Tools (Strumenti)

Servizi AWS

- AWS Directory Serviceoffre diversi modi per utilizzare Microsoft Active Directory (AD) con altri Servizi AWS come Amazon Elastic Compute Cloud (Amazon EC2), Amazon Relational Database Service (Amazon RDS) per SQL Server e FSx Amazon per Windows File Server.
- AWS Directory Service for Microsoft Active Directoryconsente ai carichi di lavoro e alle risorse compatibili con le directory di AWS utilizzare Microsoft Active Directory in. Cloud AWS
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altro modo. Account AWS
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle. Con IAM, puoi specificare chi o cosa può accedere ai servizi e alle risorse AWS, gestire centralmente le autorizzazioni granulari e analizzare l'accesso per perfezionare le autorizzazioni. AWS
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS Systems Managerti aiuta a gestire le applicazioni e l'infrastruttura in esecuzione in. Cloud AWS Semplifica la gestione delle applicazioni e delle risorse, riduce i tempi di rilevamento e risoluzione dei problemi operativi e aiuta a gestire le AWS risorse in modo sicuro su larga scala.

- AWS Systems Manager i documenti definiscono le azioni che Systems Manager esegue sulle istanze gestite. Systems Manager include più di 100 documenti preconfigurati che è possibile utilizzare specificando i parametri nel runtime (tempo di esecuzione).
- AWS Systems Manager Parameter Store è una funzionalità AWS Systems Manager e fornisce uno storage gerarchico sicuro per la gestione dei dati di configurazione e la gestione dei segreti.

Altri strumenti

- HashiCorp Terraform è uno strumento open source di infrastruttura come codice (IaC) che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud.
- PowerShellè un programma di gestione dell'automazione e della configurazione di Microsoft che funziona su Windows, Linux e macOS.
- Python è un linguaggio di programmazione per computer generico.

Deposito di codice C

Il codice per questo pattern è disponibile nell'archivio della soluzione GitHub Custom AD Cleanup Automation.

Best practice

- Unisci automaticamente i domini.Quando avvii un'istanza Windows che deve far parte di un Directory Service dominio, unisciti al dominio durante il processo di creazione dell'istanza anziché aggiungere manualmente l'istanza in un secondo momento. Per aggiungere automaticamente un dominio, seleziona la directory corretta dall'elenco a discesa Domain join directory quando avvii una nuova istanza. Per maggiori dettagli, consulta Unire senza problemi un'istanza Amazon EC2 Windows alla tua AWS Managed Microsoft AD Active Directory nella Guida all'Directory Service amministrazione.
- Eliminare gli account non utilizzati.È comune trovare account in AD che non sono mai stati utilizzati. Come gli account disabilitati o inattivi che rimangono nel sistema, gli account trascurati e inutilizzati possono rallentare il sistema AD o rendere l'organizzazione vulnerabile alle violazioni dei dati.
- Automatizza le pulizie di Active Directory.Per contribuire a mitigare i rischi per la sicurezza e impedire che gli account obsoleti influiscano sulle prestazioni di AD, è necessario eseguire le pulizie di AD a intervalli regolari. Puoi eseguire la maggior parte delle attività di gestione e pulizia

di AD scrivendo script. Le attività di esempio includono la rimozione di account disabilitati e inattivi, l'eliminazione di gruppi vuoti e inattivi e l'individuazione di account utente e password scaduti.

Poemi epici

Configurare l'ambiente

Operazione	Description	Competenze richieste
Crea una cartella di progetto e aggiungi i file.	Per clonare il repository e creare una cartella di progetto, procedi come segue: 1. Apri l'archivio di questo pattern. GitHub 2. Scegli il pulsante Codice per visualizzare le opzioni per la clonazione nel menu a discesa Clone. 3. Nella scheda HTTPS, copia l'URL fornito in Clone utilizzando l'URL web. 4. Create una cartella sul computer e denominatela con il nome del progetto. 5. Apri un terminale nel tuo computer locale e accedi a questa cartella. 6. Per clonare il repository git, usa il seguente comando. `git clone <repository-URL>.git` 7. Dopo che il repository è stato clonato, usa il seguente comando per	DevOps ingegnere

Operazione	Description	Competenze richieste
	andare alla directory clonata. `cd <directory name>` 8. Nel repository clonato, apri questo progetto in un ambiente di sviluppo integrato (IDE) di tua scelta.	

Esegui il provisioning dell'architettura di destinazione utilizzando la configurazione Terraform

Operazione	Description	Competenze richieste
Inizializza la configurazione Terraform.	Per inizializzare la directory di lavoro che contiene i file Terraform, esegui il seguente comando. `terraform init`	DevOps ingegnere
Visualizza in anteprima le modifiche.	Puoi visualizzare in anteprima le modifiche che Terraform apporterà all'infrastruttura prima che l'infrastruttura venga implementata. Per verificar e che Terraform apporti le modifiche necessarie, esegui il seguente comando. `terraform plan`	DevOps ingegnere
Esegui le azioni proposte.	Per verificare che i risultati del `terraform plan` comando siano quelli previsti, effettuate le seguenti operazioni:	DevOps ingegnere

Operazione	Description	Competenze richieste
	1. Eseguire il seguente comando seguente. `terraform apply` 2. Accedere a Console di gestione AWS e verificar e che le risorse siano presenti.	
Pulisci l'infrastruttura.	Per pulire l'infrastruttura che hai creato, usa il seguente comando. `terraform destroy` Per confermare il comando destroy, digitate`yes`.	DevOps ingegnere

Verifica della distribuzione

Operazione	Description	Competenze richieste
Esegui e testa la funzione Lambda.	Per verificare che la distribuz ione sia avvenuta correttam ente, procedi come segue: 1. Accedi a Console di gestione AWS e apri la console. Aprire la pagina Funzioni e selezionare il nome della funzione che inizia con ADcleanup-Lambda-*. 2. Nella pagina di panoramic a delle funzioni, scegliete	DevOps ingegnere

Operazione	Description	Competenze richieste
	Test nella scheda Codice nella sezione Codice sorgente. 3. Per salvare l'evento di test, fornite un nome per l'evento e scegliete Salva. Quindi, per testare l'evento, scegli di nuovo Test. I risultati dell'esecuzione mostrano l'output della funzione.	

Operazione	Description	Competenze richieste
Visualizza i risultati della funzione Lambda.	In questo modello, una EventBridge regola esegue la funzione Lambda una volta al giorno. Per visualizzare i risultati della funzione Lambda, effettuate le seguenti operazioni: 1. Accedi a Console di gestione AWS e apri la AWS Lambda console. Aprire la pagina Funzioni e selezionare il nome della funzione che inizia con ADcleanup-Lambda-*. 2. Scegliete la scheda Monitor e scegliete Visualizza registri. CloudWatch Nella CloudWatch console, la pagina Log groups mostra i risultati della funzione Lambda.	DevOps ingegnere

Pulisci l'infrastruttura dopo l'uso

Operazione	Description	Competenze richieste
Pulisci l'infrastruttura.	Per ripulire l'infrastruttura che hai creato, usa il seguente comando. `terraform destroy`	DevOps ingegnere

Operazione	Description	Competenze richieste
	Per confermare il comando destroy, digitate`yes`.	
Verifica dopo la pulizia.	Verifica che le risorse siano state rimosse correttamente.	DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
Se si tenta di rimuovere il computer AD, viene visualizzato il messaggio «Accesso negato». Il computer AD non può essere rimosso perché, per impostazione predefinita, l'azione tenta di rimuovere due indirizzi IP privati collegati come parte dei servizi AD.	Per evitare questo errore, utilizzate la seguente operazione Python per ignorare i primi due computer quando elencate le differenze tra l'output di un computer AD e l'output del computer che esegue Windows. `Difference = Difference[2:]`
Quando Lambda esegue PowerShell uno script su un server Windows, si aspetta che i moduli Active Directory siano disponibili per impostazione predefinita. Se i moduli non sono disponibili, una funzione Lambda crea un errore che indica «Get- non AdComputer è installato sull'istanza».	Per evitare questo errore, installa i moduli richiesti utilizzando i dati utente delle EC2 istanze. Usa lo EC2WindowsUserdatascript che si trova nel GitHub repository di questo pattern.

Risorse correlate

AWS documentazione

- Amazon EventBridge e AWS Identity and Access Management
- Configurazione delle autorizzazioni dell'istanza richieste per Systems Manager
- Gestione delle identità e degli accessi per Directory Service

- Aggiungi manualmente un'istanza Amazon EC2 Windows alla tua AWS Managed Microsoft AD Active Directory
- Utilizzo di politiche IAM basate sull'identità in AWS Lambda

Altre risorse

- AWS Provider (documentazione Terraform)
- Configurazione del backend (documentazione Terraform)
- Installa Terraform (documentazione Terraform)
- Modulo di avvio Python (repository Python Package Index)
- Download del file binario Terraform (documentazione Terraform)

Esegui test unitari per i lavori ETL in Python AWS Glue utilizzando il framework pytest

Praveen Kumar Jeyarajan e Vaidy Sankaran, Amazon Web Services

Riepilogo

Puoi eseguire test unitari per i lavori di estrazione, trasformazione e caricamento (ETL) in Python AWS Glue in un ambiente di sviluppo locale, ma replicare questi test in una DevOps pipeline può essere difficile e richiedere molto tempo. I test unitari possono essere particolarmente impegnativi quando si sta modernizzando il processo ETL del mainframe su stack tecnologici. AWS Questo modello mostra come semplificare i test delle unità, mantenendo intatte le funzionalità esistenti, evitando interruzioni delle funzionalità chiave delle applicazioni quando si rilasciano nuove funzionalità e mantenendo software di alta qualità. È possibile utilizzare i passaggi e gli esempi di codice in questo modello per eseguire test unitari per i lavori ETL in Python AWS Glue utilizzando il framework pytest in. AWS CodePipelineÈ inoltre possibile utilizzare questo modello per testare e distribuire più lavori. AWS Glue

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un URI di immagine di Amazon Elastic Container Registry (Amazon ECR) per AWS Glue la tua libreria, scaricato dalla Amazon ECR Public Gallery
- Terminale Bash (su qualsiasi sistema operativo) con un profilo per il target e Account AWS Regione AWS
- Python 3.10 o successivo
- Pytest
- Libreria Moto Python per i test Servizi AWS

Architecture

Il diagramma seguente descrive come incorporare i test unitari per i processi AWS Glue ETL basati su Python in una tipica pipeline di scala aziendale. AWS DevOps

Il diagramma mostra il flusso di lavoro seguente:

1. Nella fase di origine, AWS CodePipeline utilizza un bucket Amazon Simple Storage Service (Amazon S3) con versioni diverse per archiviare e gestire le risorse del codice sorgente. Queste risorse includono un esempio di Python ETL job (`sample.py`), un file di unit test (`test_sample.py`) e un modello. AWS CloudFormation Quindi, CodePipeline trasferisce il codice più recente dal ramo principale al AWS CodeBuild progetto per un'ulteriore elaborazione.
2. Nella fase di compilazione e pubblicazione, il codice più recente della fase sorgente precedente viene sottoposto a test unitari con l'aiuto di un'immagine Amazon ECR AWS Glue pubblica. Quindi, il rapporto di test viene pubblicato in gruppi di CodeBuild report. L'immagine del contenitore nell'archivio pubblico di Amazon ECR per le AWS Glue librerie include tutti i file binari necessari per eseguire le attività ETL PySparkbasate su unit test in locale. AWS Glue L'archivio pubblico del contenitore ha tre tag di immagine, uno per ogni versione supportata da. AWS Glue A scopo dimostrativo, questo modello utilizza il tag `glue_libs_4.0.0_image_01` image. Per utilizzare questa immagine contenitore come immagine di runtime in CodeBuild, copia l'URI dell'immagine che corrisponde al tag dell'immagine che intendi utilizzare, quindi aggiorna il `pipeline.yml` file nel GitHub repository per la `TestBuild` risorsa.
3. Nella fase di implementazione, il CodeBuild progetto viene avviato e pubblica il codice in un bucket Amazon S3 se tutti i test vengono superati.
4. L'utente distribuisce l' AWS Glue attività utilizzando il modello nella CloudFormation cartella. `deploy`

Tools (Strumenti)

Servizi AWS

- AWS CodeBuildè un servizio di compilazione completamente gestito che consente di compilare il codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS CodePipelineti aiuta a modellare e configurare rapidamente le diverse fasi di una versione del software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- AWS Glueè un servizio ETL completamente gestito. Ti aiuta a classificare, pulire, arricchire e spostare i dati in modo affidabile tra archivi di dati e flussi di dati.

- Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è un servizio di storage di oggetti che offre scalabilità, disponibilità dei dati, sicurezza e prestazioni all'avanguardia nel settore.

Altri strumenti

- Python è un linguaggio di programmazione generico interpretato di alto livello.
- Moto è una libreria Python per i test. Servizi AWS
- Pytest è un framework per scrivere piccoli unit test scalabili per supportare test funzionali complessi per applicazioni e librerie.
- La libreria Python ETL for AWS Glue è un repository per le librerie Python utilizzate nello sviluppo locale di processi batch per. PySpark AWS Glue

Deposito di codice

Il codice per questo pattern è disponibile nel repository GitHub aws-glue-jobs-unit-testing. Il repository include le seguenti risorse:

- Un esempio di AWS Glue job basato su Python nella cartella `src`
- Casi di unit test associati (creati utilizzando il framework pytest) nella cartella `tests`
- Un CloudFormation modello (scritto in YAML) nella cartella `deploy`

Best practice

Sicurezza per le risorse CodePipeline

È consigliabile utilizzare la crittografia e l'autenticazione per i repository di origine che si connettono alle pipeline. CodePipeline Per ulteriori informazioni, consulta le migliori pratiche di sicurezza nella CodePipeline documentazione.

Monitoraggio e registrazione delle risorse CodePipeline

È consigliabile utilizzare le funzionalità di AWS registrazione per determinare quali azioni intraprendono gli utenti nel proprio account e quali risorse utilizzano. I file di registro mostrano quanto segue:

- Ora e data delle azioni

- Indirizzo IP di origine delle azioni
- Quali azioni non sono riuscite a causa di autorizzazioni inadeguate

Le funzionalità di registrazione sono disponibili in Amazon CloudWatch Events AWS CloudTrail e Amazon. Puoi utilizzarlo CloudTrail per registrare le chiamate AWS API e gli eventi correlati effettuati da o per conto di te Account AWS. Per ulteriori informazioni, consulta la sezione Registrazione delle chiamate CodePipeline API con AWS CloudTrail nella CodePipeline documentazione.

È possibile utilizzare CloudWatch Events per monitorare le Cloud AWS risorse e le applicazioni in esecuzione su AWS. Puoi anche creare avvisi in CloudWatch Eventi. Per ulteriori informazioni, consulta Monitoraggio CodePipeline degli eventi nella CodePipeline documentazione.

Epiche

Implementa il codice sorgente

Operazione	Description	Competenze richieste

Prepara l'archivio del codice per la distribuzione.	1. Scaricalo `code.zip` dal repository GitHub aws-glue-jobs-unit-testing o crea tu stesso il file.zip utilizzando uno strumento da riga di comando. Ad esempio, puoi creare il file.zip su Linux o Mac eseguendo i seguenti comandi nel terminale:
	`git clone https://github.com/aws-samples/aws-glue-jobs-unit-testing.git`
	`cd aws-glue-jobs-unit-testing`
	`git checkout master`
	`zip -r code.zip src/ tests/ deploy/`
		DevOps ingegnere

Operazione	Description	Competenze richieste
	2. Accedi a Console di gestione AWSe scegli quello che Regione AWS preferisci. 3. Crea un bucket Amazon S3, quindi carica il pacchetto.zip e il `code.zip` file (scaricati in precedenza) nel bucket Amazon S3 che hai creato.	

Operazione	Description	Competenze richieste
Crea lo CloudFormation stack.	1. Accedi a Console di gestione AWS e quindi apri la CloudFormation console. 2. Scegli Crea stack, quindi scegli Con risorse esistenti (importa risorse). 3. Nella sezione Specifica re il modello della pagina Crea stack, scegli Carica un file modello, quindi scegli il modello pipeline.yml (scaricato dal repository). GitHub Quindi, seleziona Successivo. 4. Per il nome dello stack, inserite glue-unit-testing-pipelineo scegliete un nome di pila a vostra scelta. 5. Per ApplicationStackNa me, usa il nome glue-code pipeline-appprecompilato. Questo è il nome dello CloudFormation stack creato dalla pipeline. 6. Per BucketName, usa il nome del bucket -east-1 precompilato. aws-glue-artifacts-us Questo è il nome del bucket Amazon S3 che contiene il file.zip e viene utilizzato dalla pipeline per archiviare artefatti di codice.	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	7. Per CodeZipFile, usa il valore code.zip precompil ato. Questo è il nome chiave dell'oggetto Amazon S3 con codice di esempio. L'oggetto deve essere un file.zip. 8. Per TestReportGroupNam e, usa il nome precompilato glue-unittest-report. Questo è il nome del gruppo di rapporti di CodeBuild test creato per archiviare i report dei test unitari. 9. Scegli Avanti, quindi scegli nuovamente Avanti nella pagina Configura le opzioni dello stack. 10. Nella pagina Revisione, in Capacità, scegli l'opzione Riconosco che CloudForm ation potrebbe creare risorse IAM con nomi personalizzati. 11. Seleziona Invia. Una volta completata la creazione dello stack, puoi vedere le risorse create nella scheda Risorse. La creazione dello stack richiede circa 5-7 minuti. Lo stack crea una CodePipel ine vista utilizzando Amazon	

Operazione	Description	Competenze richieste
	S3 come origine. Nei passaggi precedenti, la pipeline è - pipeline. aws-glue-unit-test	

Esegui i test unitari

Operazione	Description	Competenze richieste
Esegui i test unitari in corso.	1. Per testare la pipeline distribuita, accedi a Console di gestione AWS, quindi apri la console. CodePipeline 2. Seleziona la pipeline creata dallo CloudFormation stack, quindi scegli Release change. La pipeline inizia a funzionare (utilizzando il codice più recente nel bucket Amazon S3). 3. Al termine della fase Test_and_build, scegli la scheda Dettagli, quindi esamina i log. 4. Scegliete la scheda Rapporti, quindi scegliete il rapporto del test dalla Cronologia report per visualizzare i risultati del test unitario. 5. Una volta completata la fase di distribuzione, esegui e monitora il AWS Glue	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	job distribuito sulla AWS Glue console. Per ulteriori informazioni, consulta Monitoraggio AWS Glue nella AWS Glue documenta zione.	

Pulisci tutte le AWS risorse

Operazione	Description	Competenze richieste
Pulisci le risorse del tuo ambiente.	Per evitare costi di infrastru ttura aggiuntivi, assicuratevi di eliminare lo stack dopo aver sperimentato gli esempi forniti in questo schema. 1. Apri la CloudFormation console, quindi seleziona lo stack che hai creato. 2. Scegli Elimina. In questo modo vengono eliminate tutte le risorse create dallo stack, inclusi i ruoli AWS Identity and Access Management (IAM), le policy IAM e i progetti. CodeBuild	AWS DevOps, DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
Il ruolo CodePipeline di servizio non può accedere al bucket Amazon S3.	• Per la policy associata al tuo ruolo CodePipeline di servizio, aggiungila `s3:ListBucket` all'elenco delle azioni della tua policy. Per istruzioni sulla visualizzazione della policy del ruolo di servizio, consulta Visualizza l'ARN della pipeline e l'ARN del ruolo di servizio (console). Modifica la dichiarazione politica per il tuo ruolo di servizio come dettagliato in Aggiungere autorizzazioni al ruolo di servizio. CodePipeline • Per la policy basata sulle risorse allegata al bucket di artifatti Amazon S3 per la tua pipeline, chiamata anche artifact bucket policy, aggiungi un'istruzione che consenta al ruolo del servizio di utilizzare l'autorizzazione. CodePipeline `s3:ListBucket`
CodePipeline restituisce un errore indicante che il bucket Amazon S3 non ha una versione.	CodePipeline richiede la versione del bucket Amazon S3 di origine. Abilita il controllo delle versioni sul tuo bucket Amazon S3. Per istruzioni, consulta Abilitare il controllo delle versioni sui bucket.

Risorse correlate

- AWS Glue
- Sviluppo e test AWS Glue di lavori a livello locale
- AWS CloudFormation per AWS Glue

Informazioni aggiuntive

Inoltre, puoi distribuire i AWS CloudFormation modelli utilizzando AWS Command Line Interface (AWS CLI). Per ulteriori informazioni, consulta Distribuzione rapida di modelli con trasformazioni nella documentazione. CloudFormation

Configura una CI/CD pipeline utilizzando AWS CodePipeline e AWS CDK

Konstantin Zarudaev, Yasha Dabas, Lars Kinder e Cizer Pereira, Amazon Web Services

Home

L'automazione del processo di creazione e rilascio del software con integrazione e distribuzione continue (CI/CD) supports repeatable builds and rapid delivery of new features to your users. You can quickly and easily test each code change, and you can catch and fix bugs before releasing your software. By running each change through your staging and release process, you can verify the quality of your application or infrastructure code. CI/CDincorpora una cultura, una serie di principi operativi e una raccolta di pratiche che aiutano i team di sviluppo delle applicazioni a fornire modifiche al codice con maggiore frequenza e affidabilità). L'implementazione è anche nota come pipeline CI/CD.

Questo modello definisce una pipeline riutilizzabile di integrazione continua e distribuzione continua (CI/CD) su Amazon Web Services (AWS) con un repository AWS. CodeCommit La CodePipeline pipeline AWS è scritta utilizzando AWS Cloud Development Kit (AWS CDK) v2.

Utilizzando CodePipeline, puoi modellare le diverse fasi del processo di rilascio del software tramite l'interfaccia della Console di gestione AWS, l'AWS Command Line Interface (AWS CLI), AWS o CloudFormation AWS. SDKs Questo modello dimostra l'implementazione CodePipeline e i relativi componenti utilizzando AWS CDK. Oltre alle librerie di costruzione, AWS CDK include un toolkit (il comando CLI`cdk`), che è lo strumento principale per interagire con l'app AWS CDK. Tra le altre funzioni, il toolkit offre la possibilità di convertire uno o più stack in CloudFormation modelli e distribuirli su un account AWS.

La pipeline include test per convalidare la sicurezza delle librerie di terze parti e aiuta a garantire un rilascio rapido e automatico negli ambienti specificati. È possibile aumentare la sicurezza complessiva delle applicazioni sottoponendole a un processo di convalida.

Lo scopo di questo modello è accelerare l'uso delle CI/CD pipeline per distribuire il codice, garantendo al contempo che le risorse distribuite rispettino le migliori pratiche. DevOps Dopo aver implementato il codice di esempio, avrai un AWS CodePipeline con processi di linting, test, controllo di sicurezza, implementazione e post-distribuzione. Questo modello include anche i passaggi per Makefile. Utilizzando un Makefile, gli sviluppatori possono riprodurre CI/CD i passaggi localmente e aumentare la velocità del processo di sviluppo.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Una conoscenza di base di quanto segue:
  - AWS CDK
  - AWS CloudFormation
  - AWS CodePipeline
  - TypeScript

Limitazioni

Questo modello utilizza AWS CDK TypeScript solo per. Non copre altre lingue supportate da AWS CDK.

Versioni del prodotto

Utilizza le versioni più recenti dei seguenti strumenti:

- Interfaccia a riga di comando di AWS (CLI AWS)
- cfn_nag
- git-remote-codecommit
- Node.js

Architecture

Stack tecnologico Target

- AWS CDK
- AWS CloudFormation
- AWS CodeCommit
- AWS CodePipeline

Architettura Target

La pipeline viene attivata da una modifica nel CodeCommit repository AWS (). `SampleRepository` All'inizio, CodePipeline crea artefatti, si aggiorna e avvia il processo di distribuzione. La pipeline risultante implementa una soluzione in tre ambienti indipendenti:

- Dev: controllo del codice in tre fasi nell'ambiente di sviluppo attivo
- Test: ambiente di test di integrazione e regressione
- Prod — Ambiente di produzione

I tre passaggi inclusi nella fase di sviluppo sono il linting, la sicurezza e i test unitari. Questi passaggi vengono eseguiti in parallelo per accelerare il processo. Per garantire che la pipeline fornisca solo artefatti funzionanti, verrà interrotta ogni volta che una fase del processo fallisce. Dopo una fase di implementazione in fase di sviluppo, la pipeline esegue test di convalida per verificare i risultati. In caso di successo, la pipeline distribuirà quindi gli artefatti nell'ambiente di test, che contiene la convalida post-implementazione. Il passaggio finale consiste nel distribuire gli artefatti nell'ambiente Prod.

Il diagramma seguente mostra il flusso di lavoro dal CodeCommit repository ai processi di compilazione e aggiornamento eseguiti da CodePipeline, le tre fasi dell'ambiente di sviluppo e la successiva implementazione e convalida in ciascuno dei tre ambienti.

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software che aiuta a definire e fornire l'infrastruttura cloud AWS in codice.
- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account e regioni AWS. In questo modello CloudFormation i modelli possono essere utilizzati per creare un CodeCommit repository e una CodePipeline pipeline CI/CD.
- AWS CodeCommit è un servizio di controllo delle versioni che consente di archiviare e gestire archivi Git in modo privato, senza dover gestire il proprio sistema di controllo del codice sorgente.
- AWS CodePipeline è un CI/CD servizio che ti aiuta a modellare e configurare rapidamente le diverse fasi di un rilascio di software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.

- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.

Altri strumenti

- cfn_nag è uno strumento open source che cerca modelli nei CloudFormation modelli per identificare potenziali problemi di sicurezza.
- git-remote-codecommitè un'utilità per inviare ed estrarre codice dai CodeCommit repository estendendo Git.
- Node.js è un ambiente di JavaScript runtime basato sugli eventi progettato per la creazione di applicazioni di rete scalabili.

Codice

Il codice per questo modello è disponibile nel repository GitHub AWS CodePipeline with CI/CD practices.

Best practice

Esamina le risorse, come le policy di AWS Identity and Access Management (IAM), per confermare che siano in linea con le best practice della tua organizzazione.

Epiche

Installa strumenti

Operazione	Description	Competenze richieste

Installa strumenti su macOS o Linux.	Se utilizzi macOS o Linux, puoi installare gli strumenti eseguendo il seguente comando nel tuo terminale preferito o usando Homebrew per Linux.
	`brew install`
	`brew install git-remot`
	`e-codecommit`
		DevOps ingegnere

Operazione	Description	Competenze richieste
	`brew install ruby brew-gem` `brew-gem install cfn-nag`	
Configura AWS CLI.	Per configurare AWS CLI, usa le istruzioni per il tuo sistema operativo: • Windows: passaggi di configurazione per le connessioni HTTPS ai CodeCommit repository AWS su Windows con l'helper di credenziali AWS CLI • Linux, macOS, Unix: passaggi di configurazione per le connessioni HTTPS ai CodeCommit repository AWS su Linux, macOS o Unix con l'helper di credenziali AWS CLI	DevOps ingegnere

Configura la distribuzione iniziale

Operazione	Description	Competenze richieste
Scarica o clona il codice.	Per ottenere il codice utilizzato da questo pattern, effettuate una delle seguenti operazioni: • Scaricate il codice sorgente più recente dalle versioni del GitHub repository	DevOps ingegnere

Operazione	Description	Competenze richieste

	e decomprimete il file scaricato in una cartella.
	• Clona il progetto eseguendo il seguente comando.
	
	`git clone --depth 1 https://github.com/aws-samples/aws-codepipeline-cicd.git`
	
	Rimuovi la `.git` directory dal repository clonato.
	
	`cd ./aws-codepipeline-cicd`
	`rm -rf ./.git`
	
	Successivamente, utilizzer ai un CodeCommit repositor y AWS appena creato come origine remota.	

Operazione	Description	Competenze richieste

Connect all'account AWS.	Puoi connetterti utilizzando un token di sicurezza temporaneo o l'autenticazione delle landing zone. Per confermare che stai utilizzando l'account e la regione AWS corretti, esegui i seguenti comandi.
	`AWS_REGION="eu-west-1"`
	`ACCOUNT_NUMBER=$(aws sts get-caller-identity --query Account --output text)`
	`echo "${ACCOUNT_NUMBER}"`
		DevOps ingegnere

Operazione	Description	Competenze richieste

Avvia l'ambiente.	Per avviare un ambiente AWS CDK, esegui i seguenti comandi.
	`npm install`
	`npm run cdk bootstrap "aws://${ACCOUNT_NUMBER}/${AWS_REGION}"`
	Dopo aver avviato correttamente l'ambiente, dovrebbe essere visualizzato il seguente output.
	`# Bootstrapping environment aws://{account}/{region}...`
	`# Environment aws://{account}/{region} bootstrapped`
	Per ulteriori informazioni sul bootstrap di AWS CDK, consulta la documentazione di AWS CDK.
		DevOps ingegnere

Operazione	Description	Competenze richieste

Sintetizza un modello.	Per sintetizzare un'app AWS CDK, usa il comando. `cdk synth`
	`npm run cdk synth`
	Vedrai il seguente output.
	`Successfully synthesized to <path-to-directory>/aws-codepipeline-cicd/cdk.out`
	`Supply a stack id (CodePipeline, Dev-MainStack) to display its template.`
		DevOps ingegnere

Operazione	Description	Competenze richieste

Distribuisci lo CodePipeline stack.	Ora che hai avviato e sintetizzato il CloudFormation modello, puoi distribuirlo. La distribuzione creerà la CodePipeline pipeline e un CodeCommit repository, che saranno l'origine e il trigger della pipeline.
	`npm run cdk -- deploy CodePipeline --require-approval never`
	Dopo aver eseguito il comando, dovresti vedere una corretta distribuzione dello CodePipeline stack e delle informazioni di output. ti `CodePipeline.RepositoryName` dà il nome del CodeCommit repository nell'account AWS.
	`CodePipeline: deploying...`
	`CodePipeline: creating CloudFormation changeset...`
	`# CodePipeline Outputs:`
	`CodePipeline.RepositoryName = SampleRepository`
	`Stack ARN: arn:aws:cloudformation:REGION:ACCOUNT-ID`
		DevOps ingegnere

Operazione	Description	Competenze richieste
	`:stack/CodePipeline/STACK-ID`	

Operazione	Description	Competenze richieste

Configura l' CodeCommit archivio e la filiale remoti.	Dopo una distribuzione di successo, CodePipeline avvierà la prima esecuzione della pipeline, che puoi trovare nella console CodePipeline AWS. Poiché AWS CDK e io CodeCommit non avviamo un ramo predefinito, questa esecuzione iniziale della pipeline fallirà e restituirà il seguente messaggio di errore.
	`The action failed because no branch named main was found in the selected AWS CodeCommit repository SampleRepository. Make sure you are using the correct branch name, and then try again. Error: null`
	Per correggere questo errore, configura un'origine remota come `SampleRepository` e crea il ramo richiesto. `main`
	`RepoName=$(aws cloudformation describe-stacks --stack-name CodePipeline --query "Stacks[0].Outputs[?OutputKey=='RepositoryName'].OutputValue" --output text)`
	`echo "${RepoName}"`
	`#`
		DevOps ingegnere

Operazione	Description	Competenze richieste

	`git init`
	`git branch -m master main`
	`git remote add origin codecommit://${RepoName}`
	`git add .`
	`git commit -m "Initial commit"`
	`git push -u origin main`	

Testa la pipeline implementata CodePipeline

Operazione	Description	Competenze richieste
Apporta una modifica per attivare la pipeline.	Dopo una corretta implement azione iniziale, dovresti disporre di una CI/CD pipeline completa con un `main` ramo `SampleRepository` come ramo di origine. Non appena confermate le modifiche al `main` ramo, la pipeline avvierà ed eseguirà la seguente sequenza di azioni: 1. Recupera il codice dal repository. CodeCommit 2. Crea il tuo codice. 3. Aggiorna la pipeline stessa (`UpdatePipeline`). 4. Esegui tre job paralleli per i controlli di linting, sicurezza e unit test.	DevOps ingegnere

Operazione	Description	Competenze richieste
	5. In caso di successo, la pipeline distribuirà lo `Main` stack dall'`./lib/main-stack.ts` ambiente Dev. 6. Esegui un controllo post-implementazione per le risorse distribuite. Puoi seguire tutti i CodePipeline passaggi e i risultati nella CodePipeline console. 7. In caso di successo, la pipeline ripeterà l'impleme ntazione e la convalida per gli ambienti Test e Prod.	

Esegui il test localmente utilizzando un Makefile

Operazione	Description	Competenze richieste
Esegui il processo di sviluppo utilizzando un Makefile.	È possibile eseguire l'intera pipeline localmente utilizzan do il `make` comando oppure eseguire un singolo passaggio (ad esempio,`make linting`). Per testare l'utilizzo`make`, effettuate le seguenti azioni: • Implementa la pipeline locale: `make` • Esegui solo test unitari: `make unittest`	Sviluppatore di app, DevOps ingegnere

Operazione	Description	Competenze richieste
	• Esegui la distribuzione sull'account corrente: `make deploy` • Pulisci l'ambiente: `make clean`	

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Elimina le risorse dell'app AWS CDK.	Per pulire l'app AWS CDK, esegui il comando seguente. `cdk destroy --all` Tieni presente che i bucket Amazon Simple Storage Service (Amazon S3) creati durante il bootstrap non vengono eliminati automatic amente. Hanno bisogno di una policy di conservaz ione che consenta l'elimina zione, oppure devi eliminarli manualmente nel tuo account AWS.	DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
Il modello non funziona come previsto.	Se qualcosa va storto e il modello non funziona, assicurati di avere quanto segue:

Problema	Soluzione
	• Le versioni corrette degli strumenti. • Accesso all'account AWS di destinazione (connettività di rete). • Autorizzazioni sufficienti per l'account AWS di destinazione.

Risorse correlate

- Nozioni di base delle attività comuni nel Centro identità IAM
- CodePipeline Documentazione AWS
- CDK AWS

Configura la registrazione centralizzata su scala aziendale utilizzando Terraform

Aarti Rajput, Yashwant Patel e Nishtha Yadav, Amazon Web Services

Riepilogo

La registrazione centralizzata è fondamentale per l'infrastruttura cloud di un'organizzazione, perché offre visibilità sulle operazioni, sulla sicurezza e sulla conformità. Man mano che l'organizzazione ridimensiona il proprio AWS ambiente su più account, una strategia strutturata di gestione dei log diventa fondamentale per eseguire operazioni di sicurezza, soddisfare i requisiti di audit e raggiungere l'eccellenza operativa.

Questo modello fornisce un framework scalabile Account AWS e sicuro per centralizzare i log di più servizi, per consentire una gestione dei log su scala aziendale in implementazioni complesse. AWS La soluzione è automatizzata utilizzando Terraform, uno strumento di infrastruttura come codice (IaC) che garantisce implementazioni coerenti e ripetibili e riduce al minimo la configurazione manuale. HashiCorp Combinando Amazon CloudWatch Logs, Amazon Data Firehose e Amazon Simple Storage Service (Amazon S3), puoi implementare una solida pipeline di aggregazione e analisi dei log che offre:

- Gestione centralizzata dei log in tutta l'organizzazione in AWS Organizations
- Raccolta automatizzata dei log con controlli di sicurezza integrati
- Elaborazione scalabile dei log e archiviazione durevole
- Report di conformità e audit trail semplificati
- Informazioni operative e monitoraggio in tempo reale

La soluzione raccoglie i log dai contenitori, dalle funzioni e dalle istanze di database Amazon Elastic Kubernetes Service (Amazon EKS AWS Lambda) tramite Logs. CloudWatch Inoltra automaticamente questi log a un account di registrazione dedicato utilizzando i filtri di abbonamento. CloudWatch Firehose gestisce la pipeline di streaming di log ad alta velocità verso Amazon S3 per lo storage a lungo termine. Amazon Simple Queue Service (Amazon SQS) è configurato per ricevere notifiche di eventi di Amazon S3 al momento della creazione di oggetti. Ciò consente l'integrazione con i servizi di analisi, tra cui:

- Amazon OpenSearch Service per la ricerca, la visualizzazione e l'analisi dei log in tempo reale

- Amazon Athena per interrogazioni basate su SQL
- Amazon EMR per l'elaborazione su larga scala
- Lambda per la trasformazione personalizzata
- Amazon Quick Sight per dashboard

Tutti i dati vengono crittografati utilizzando AWS Key Management Service (AWS KMS) e l'intera infrastruttura viene distribuita utilizzando Terraform per una configurazione coerente in tutti gli ambienti.

Questo approccio di registrazione centralizzato consente alle organizzazioni di migliorare il proprio livello di sicurezza, mantenere i requisiti di conformità e ottimizzare l'efficienza operativa nell'intera infrastruttura. AWS

Prerequisiti e limitazioni

Prerequisiti

- Una landing zone per la tua organizzazione costruita utilizzando AWS Control Tower
- Account Factory for Terraform (AFT), distribuito e configurato con gli account richiesti
- Terraform per il provisioning dell'infrastruttura
- AWS Identity and Access Management Ruoli e politiche (IAM) per l'accesso tra account diversi

Per istruzioni sulla configurazione AWS Control Tower degli account AFT e Application, consulta la sezione Epics.

Account richiesti

La tua organizzazione AWS Organizations dovrebbe includere questi account:

- Account dell'applicazione: uno o più account di origine in cui Servizi AWS (Amazon EKS, Lambda e Amazon RDS) eseguono e generano log
- Account Log Archive: un account dedicato per l'archiviazione e la gestione centralizzate dei log

Versioni del prodotto

- AWS Control Tower versione 3.1 o successiva

- Terraform versione 0.15.0 o successiva

Architecture

Il diagramma seguente illustra un'architettura di registrazione AWS centralizzata che fornisce una soluzione scalabile per la raccolta, l'elaborazione e l'archiviazione dei log da più account di applicazioni in un account Log Archive dedicato. Questa architettura gestisce in modo efficiente i log provenienti da Servizi AWS, tra cui Amazon RDS, Amazon EKS e Lambda, e li indirizza attraverso un processo semplificato ai bucket S3 regionali nell'account Log Archive.

Il flusso di lavoro include cinque processi:

1. Processo di flusso di log
   - Il processo di flusso di log inizia negli account dell'applicazione, dove vengono Servizi AWS generati vari tipi di log, ad esempio log generali, di errore, di controllo, di query lente da Amazon RDS, log del piano di controllo da Amazon EKS e log di esecuzione di funzioni ed errori da Lambda.
   - CloudWatch funge da punto di raccolta iniziale. Raccoglie questi registri a livello di gruppo di log all'interno di ciascun account dell'applicazione.
   - In CloudWatch, i filtri di abbonamento determinano quali registri devono essere inoltrati all'account centrale. Questi filtri offrono un controllo granulare sull'inoltro dei log, in modo da poter specificare modelli di log esatti o flussi di log completi per la centralizzazione.
2. Trasferimento dei log tra account
   - I log vengono spostati nell'account Log Archive. CloudWatch i filtri di abbonamento facilitano il trasferimento tra account e preservano il contesto regionale.
   - L'architettura stabilisce più flussi paralleli per gestire diverse fonti di log in modo efficiente, per garantire prestazioni e scalabilità ottimali.
3. Elaborazione dei log nell'account Log Archive
   - Nell'account Log Archive, Firehose elabora i flussi di log in entrata.
   - Ogni regione mantiene flussi di distribuzione Firehose dedicati in grado di trasformare, convertire o arricchire i log in base alle esigenze.
   - Questi stream Firehose inviano i log elaborati ai bucket S3 nell'account Log Archive, che si trova nella stessa regione degli account dell'applicazione di origine (Regione A nel diagramma) per mantenere i requisiti di sovranità dei dati.

4. Notifiche e flussi di lavoro aggiuntivi
   - Quando i log raggiungono i bucket S3 di destinazione, l'architettura implementa un sistema di notifica utilizzando Amazon SQS.
   - Le code SQS regionali consentono l'elaborazione asincrona e possono attivare flussi di lavoro, analisi o sistemi di avviso aggiuntivi in base ai log archiviati.
5. AWS KMS per motivi di sicurezza

   L'architettura incorpora funzionalità AWS KMS di sicurezza. AWS KMS fornisce chiavi di crittografia per i bucket S3. Ciò garantisce che tutti i log archiviati mantengano la crittografia inattiva, mantenendo al contempo la crittografia regionale per soddisfare i requisiti di residenza dei dati.

Tools (Strumenti)

Servizi AWS

- Amazon CloudWatch è un servizio di monitoraggio e osservabilità che raccoglie dati operativi e di monitoraggio sotto forma di log, metriche ed eventi. Fornisce una visione unificata di AWS risorse, applicazioni e servizi eseguiti su AWS e server locali.
- CloudWatch I filtri di sottoscrizione dei log sono espressioni che corrispondono a uno schema negli eventi di registro in entrata e forniscono gli eventi di log corrispondenti alla AWS risorsa specificata per ulteriori elaborazioni o analisi.
- AWS Control Tower Account Factory For Terraform (AFT) imposta una pipeline Terraform per aiutarti a fornire e personalizzare gli account. AWS Control Tower AFT fornisce il provisioning degli account basato su Terraform, consentendoti al contempo di gestire i tuoi account con. AWS Control Tower
- Amazon Data Firehose fornisce dati di streaming in tempo reale verso destinazioni come Amazon S3, Amazon Redshift e Amazon Service. OpenSearch Si ridimensiona automaticamente in base alla velocità di trasmissione dei dati e non richiede alcuna amministrazione continua.
- Amazon Elastic Kubernetes Service (Amazon EKS) è un servizio di orchestrazione di container gestito che semplifica la distribuzione, la gestione e la scalabilità delle applicazioni containerizzate utilizzando Kubernetes. Gestisce automaticamente la disponibilità e la scalabilità dei nodi del piano di controllo Kubernetes.
- AWS Key Management Service (AWS KMS) crea e controlla le chiavi di crittografia per crittografare i dati. AWS KMS si integra con altri Servizi AWS per aiutarti a proteggere i dati archiviati con questi servizi.

- AWS Lambdaè un servizio di elaborazione serverless che consente di eseguire codice senza fornire o gestire server. Ridimensiona automaticamente le applicazioni eseguendo il codice in risposta a ciascun trigger e addebita solo in base al tempo di calcolo utilizzato.
- Amazon Relational Database Service (Amazon RDS) è un servizio di database relazionale gestito che semplifica la configurazione, il funzionamento e la scalabilità di un database relazionale nel cloud. Fornisce una capacità ridimensionabile e conveniente, automatizzando al contempo attività amministrative dispendiose in termini di tempo.
- Amazon Simple Queue Service (Amazon SQS) è un servizio di accodamento messaggi che consente di disaccoppiare e scalare microservizi, sistemi distribuiti e applicazioni serverless. Elimina la complessità della gestione e del funzionamento del middleware orientato ai messaggi.
- Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è un servizio di storage di oggetti basato sul cloud che offre scalabilità, disponibilità dei dati, sicurezza e prestazioni. Può archiviare e recuperare qualsiasi quantità di dati da qualsiasi punto del Web.

Altri strumenti

- Terraform è uno strumento di infrastruttura come codice (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.

Codice

Il codice per questo modello è disponibile nel repository di registrazione GitHub centralizzato.

Best practice

- Utilizzane più Account AWS in una singola organizzazione in. AWS Organizations Questa pratica consente la gestione centralizzata e la registrazione standardizzata tra gli account.
- Configura i bucket S3 con controllo delle versioni, politiche del ciclo di vita e replica tra regioni. Implementa la crittografia e la registrazione degli accessi per la sicurezza e la conformità.
- Implementa standard di registrazione comuni utilizzando il formato JSON con timestamp e campi standard. Utilizza una struttura e una correlazione di prefissi coerenti per semplificare il monitoraggio e l'analisi. IDs
- Abilita i controlli di sicurezza con AWS KMS crittografia e accesso con privilegi minimi. Mantieni AWS CloudTrail il monitoraggio e la rotazione regolare delle chiavi per una maggiore sicurezza.
- Imposta CloudWatch metriche e avvisi per il monitoraggio della consegna. Monitora costi e prestazioni con notifiche automatiche.

- Configura le politiche di conservazione di Amazon S3 per soddisfare i requisiti di conformità e abilita la registrazione degli accessi ai server Amazon S3 per tenere traccia di tutte le richieste effettuate ai tuoi bucket S3. Conserva la documentazione per le politiche e le regole del ciclo di vita dei bucket S3. Esegui revisioni periodiche dei log di accesso, delle autorizzazioni dei bucket e delle configurazioni di archiviazione per garantire le migliori pratiche di conformità e sicurezza.

Epiche

Configura account AWS Control Tower, AFT e Application

Operazione	Description	Competenze richieste
Configura un AWS Control Tower ambiente con AFT.	1. Esegui la distribuzione AWS Control Tower seguendo le istruzioni contenute nella AWS Control Tower documentazione. 2. Implementate AFT seguendo le istruzioni contenute nella AWS Control Tower documentazione.	Amministratore AWS
Abilita la condivisione delle risorse per l'organizzazione.	1. Configura AWS Command Line Interface (AWS CLI) con le credenziali dell'account di gestione, che forniscono le autorizzazioni amministrative per la gestione. AWS Control Tower 2. Esegui il seguente AWS CLI comando in qualsiasi momento: Regione AWS	Amministratore AWS

Operazione	Description	Competenze richieste
	`aws ram enable-sharing-with-aws-organization` Ciò consente la condivisione delle risorse all'interno dell'organizzazione in AWS Organizations tutte le regioni che supportano AWS Resource Access Manager (AWS RAM).	
Verifica o fornisci gli account delle applicazioni.	Per fornire nuovi account dell'Applicazione per il tuo caso d'uso, creali tramite AFT. Per ulteriori informazioni, consulta Fornire un nuovo account con AFT nella AWS Control Tower documentazione.	Amministratore AWS

Configura i file di configurazione per gli account dell'applicazione

Operazione	Description	Competenze richieste
Copia il contenuto della `Application_account` cartella nel `aft-account-customizations` repository.	1. Crea una cartella denominata `Application_account` nel percorso principale del `aft-account-customizations` repository. Questo repository viene creato automaticamente	DevOps ingegnere

Operazione	Description	Competenze richieste
	quando configuri AFT (vedi l'epopea precedente). 2. Passate alla directory principale del scale-using-terraform repository centralised-logging-at-enterprise-, copiate il contenuto della `aft/account` directory e incollatelo nella `Application_account` directory che avete creato nel passaggio 1 del repository. `aft-account-customizations` 3. Dalla directory principale del `centralised-logging-at-enterprise-scale-using-terraform` repository, copia il contenuto della `Application_account` directory nella `Application_account/terraform` directory del repository. `aft-account-customizations` 4. Nel `aft-account-customizations/Application_account/terraform.tfvars` file, conferma che tutti i parametri vengano passati come argomenti nei file di	

Operazione	Description	Competenze richieste
	configurazione Terraform corrispondenti.	

Operazione	Description	Competenze richieste
Rivedi e modifica i parametri di input per la configurazione dell'account dell'applicazione.	In questo passaggio, configuri il file di configurazione per la creazione di risorse negli account dell'applicazione, inclusi gruppi di CloudWatch log, filtri di CloudWatch sottoscrizione, ruoli e policy IAM e dettagli di configurazione per le funzioni Amazon RDS, Amazon EKS e Lambda. Nel tuo `aft-account-customizations` repository, nella `Application_account` cartella, configura i parametri di input nel `terraform.tfvars` file in base ai requisiti della tua organizzazione: • `environment` : il nome dell'ambiente (ad esempio, `proddev`,`staging`) in cui verranno distribuite le risorse. • `account_name` : Il nome del Account AWS luogo in cui verranno create le risorse. • `log_archive_account_id` : L' Account AWS ID in cui verranno archiviati i log.	DevOps ingegnere

Operazione	Description	Competenze richieste
	• `admin_role_name` : il nome del ruolo amministrativo che verrà utilizzato per la gestione delle risorse. • `tags`: Una mappa di coppie chiave-valore che rappresentano tag comuni da applicare a tutte le risorse. • `rds_config` : un oggetto che contiene i dettagli di configurazione per le istanze Amazon RDS. • `allowed_cidr_blocks` : Un elenco di blocchi CIDR a cui è consentito accedere alle risorse. • `destination_name:` Una variabile utilizzata per creare l'Amazon Resource Name (ARN) della CloudWatch destinazione in cui verranno trasmessi i log. • `rds_parameters:` Un oggetto che contiene le impostazioni del gruppo di parametri Amazon RDS. • `vpc_config` : un oggetto che contiene i dettagli di configurazione del VPC. • `eks_config` : un oggetto che contiene i dettagli di	

Operazione	Description	Competenze richieste
	configurazione per i cluster Amazon EKS. • `lambda_config` : un oggetto che contiene i dettagli di configurazione per le funzioni Lambda. • `restrictive_cidr_range` : Un elenco di intervalli CIDR restrittivi per le regole dei gruppi di sicurezza. • `target_account_id` : L'Account AWS ID dell'account Log Archive di destinazione in cui verranno distribuite le risorse.	

Imposta i file di configurazione per l'account Log Archive

Operazione	Description	Competenze richieste
Copia il contenuto della `Log_archive_account` cartella nel `aft-account-customizations` repository.	1. Crea una cartella denominata `Log_archive_account` nel percorso principale del `aft-account-customizations` repository. Questo repository viene creato automaticamente quando configuri AFT. 2. Passate alla directory principale del `centralised-logging-at-ente`	DevOps ingegnere

Operazione	Description	Competenze richieste
	`rprise-scale-using-terraform` repository, copiate il contenuto della `aft/account` directory e incollatelo nella `Log_archive_account` directory creata nel passaggio precedente del repository. `aft-account-customizations` 3. Dalla directory principale del `centralised-logging-at-enterprise-scale-using-terraform` repository, copia il contenuto della `Log_archive_account` directory nella `Log_archive_account/terraform` directory del repository. `aft-account-customizations` 4. Nel `aft-account-customizations/Log_archive_account/terraform.tfvars` file, conferma che tutti i parametri vengano passati come argomenti nei file di configurazione Terraform corrispondenti.	

Operazione	Description	Competenze richieste
Rivedi e modifica i parametri di input per la configurazione dell'account Log Archive.	In questo passaggio, configuri il file di configurazione per la creazione di risorse nell'account Log Archive, inclusi i flussi di distribuzione Firehose, i bucket S3, le code SQS e i ruoli e le policy IAM. Nella `Log_archive_account` cartella del tuo `aft-account-customizations` repository, configura i parametri di input nel `terraform.tfvars` file in base ai requisiti della tua organizzazione: • `environment` : il nome dell'ambiente (ad esempio, `proddev`,`staging`) in cui verranno distribuite le risorse. • `destination_name` : variabile utilizzata per creare l'ARN della CloudWatch destinazione in cui verranno trasmessi i log. • `source_account_ids` : Un elenco di quelle a Account AWS IDs cui è consentito inserire filtri di sottoscrizione nella destinazione del registro. Puoi inserire tutti gli account IDs che desideri abilitare	DevOps ingegnere

Operazione	Description	Competenze richieste
	per la registrazione centralizzata.	

Esegui i comandi Terraform per fornire risorse

Operazione	Description	Competenze richieste

Opzione 1: distribuisci i file di configurazione Terraform da AFT.	In AFT, la pipeline AFT viene attivata dopo aver inviato il codice con le modifiche alla configurazione nel repository. GitHub `aft-account-customizations` AFT rileva automaticamente le modifiche e avvia il processo di personalizzazione dell'account.
	
	Dopo aver apportato modifiche ai file Terraform (`terraform.tfvars`), esegui il commit e invia le modifiche al tuo repository: `aft-account-customizations`
	
	`$ git add *`
	`$ git commit -m "update message"`
	`$ git push origin main`
	
	Note
	Se stai usando un ramo diverso (ad esempio`dev`), sostituis
		DevOps ingegnere

Operazione	Description	Competenze richieste
	cilo `main` con il nome del ramo.	

Operazione	Description	Competenze richieste

Opzione 2: distribuisci manualmente il file di configurazione Terraform.	Se non utilizzi AFT o desideri distribuire la soluzione manualmente, puoi utilizzare i seguenti comandi Terraform dalle cartelle and: `Application_account` `Log_archive_account`
	1. Clona il GitHub repository e configura i parametri di input nel file. `terraform.tfvars`
	2. Esegui il comando seguente:
	`$ terraform init`
	3. Anteprima delle modifiche:
	`$ terraform plan`
	Questo comando valuta la configurazione di Terraform per determinare lo stato desiderato delle risorse e la confronta con lo stato attuale dell'infrastruttura.
	4. Applica le modifiche:
	`$ terraform apply`
	5. Rivedi le modifiche pianificate e digita sì quando richiesto per procedere con l'applicazione.
		DevOps ingegnere

Convalida le risorse

Operazione	Description	Competenze richieste
Verifica i filtri di abbonamento.	Per verificare che i filtri di abbonamento inoltrino correttamente i log dai gruppi di log degli account dell'appl icazione all'account Log Archive: 1. Nell'account dell'appl icazione, apri la CloudWatc h console. 2. Nel pannello di navigazione a sinistra, scegli Log groups (Gruppi di log). 3. Seleziona ogni gruppo di log (`/aws/rds`,`/aws/ eks`,`/aws/lambda`) e scegli la scheda Filtri di abbonamento. Dovresti vedere filtri di abbonamento attivi che puntano all'ARN di destinazione, in base al nome specificato nel file di configurazione Terraform. 4. Scegli un filtro di abbonamento per verificar ne la configurazione e lo stato.	DevOps ingegnere
Verifica i flussi Firehose.	Per verificare che i flussi Firehose nell'account Log	DevOps ingegnere

Operazione	Description	Competenze richieste
	Archive elaborino correttam ente i log dell'applicazione: 1. Nell'account Log Archive, apri la console Firehose. 2. Nel riquadro di navigazione a sinistra, scegli Firehose stream. 3. Scegli uno stream Firehose e verifica quanto segue: • La destinazione mostra il bucket S3 corretto. • La scheda Monitorag gio mostra le metriche di consegna corrette. • L'orario di consegna recente è aggiornato.	

Operazione	Description	Competenze richieste
Convalida i bucket S3 centralizzati.	Per verificare che i bucket S3 centralizzati ricevano e organizzino correttamente i log: 1. Nell'account Log Archive, apri la console Amazon S3. 2. Seleziona ogni bucket di registrazione centrale. 3. Naviga attraverso la struttura delle cartelle: . AWSLogs/AccountID/Region/Service Dovresti vedere i file di registro organizzati per timestamp () YYYY/MM/DD/HH. 4. Scegli un file di registro recente e verificane il formato e l'integrità dei dati.	DevOps ingegnere

Operazione	Description	Competenze richieste
Convalida le code SQS.	Per verificare che le code SQS ricevano notifiche per i nuovi file di registro: 1. Nell'account Log Archive, apri la console Amazon SQS. 2. Nel riquadro di navigazione sinistro scegliere Queues (Code). 3. Seleziona ogni coda configurata e scegli Invia e ricevi messaggi. Dovresti vedere messaggi che contengono notifiche di eventi S3 per i nuovi file di registro. 4. Scegli un messaggio per verificare che contenga le informazioni corrette sull'oggetto S3.	DevOps ingegnere

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Opzione 1: rimuovere il file di configurazione Terraform da AFT.	Quando rimuovi i file di configurazione Terraform e inserisci le modifiche, AFT avvia automaticamente il processo di rimozione delle risorse.	DevOps ingegnere

Operazione	Description	Competenze richieste

	1. Vai al repository. `aft-account-customizations`
	2. Vai alla directory `terraform` .
	3. Eliminare i seguenti file:
	• `modules`directory
	• `iam.tf`
	• `versions.tf`
	• `variables.tf`
	• `outputs.tf`
	• `terraform.tfvars`
	4. Cancella il contenuto del `main.tf` file.
	5. Invia le modifiche al repository:
	`# Stage all changes`
	`$ git add *`
	`# Commit cleanup changes`
	`$ git commit -m "Remove AFT customizations"`
	`# Push to repository`
	`$ git push origin main`
	Note
	Se utilizzi un ramo diverso (ad esempio`dev`),	

Operazione	Description	Competenze richieste
	sostituiscilo `main` con il nome del ramo.	

Operazione	Description	Competenze richieste

Opzione 2: ripulisci manualmente le risorse Terraform.	Se non utilizzi AFT o desideri pulire le risorse manualmente, usa i seguenti comandi Terraform dalle cartelle `Application_account` and`Log_archive_account` :
	1. Inizializza la configurazione Terraform:
	`$ terraform init`
	Questo comando inizializza Terraform e garantisce l'accesso allo stato corrente.
	2. Anteprima delle modifiche alla pulizia:
	`$ terraform destroy`
	Questo comando valuta quali risorse verranno distrutte e confronta lo stato desiderato con lo stato attuale dell'infrastruttura.
	3. Esegui la pulizia. Quando richiesto, digita sì per confermare ed eseguire il piano di distruzione.
		DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
La destinazione CloudWatch Logs non è stata creata o è inattiva.	Convalida quanto segue: 1. Nell'account Log Archive, verifica che la politica di destinazione includa: • L'account principale di origine corretto. • L'azione corretta (`logs:PutSubscriptionFilter`). • Un ARN di destinazione valido. 2. Verificate che lo stream Firehose esista e sia attivo. 3. Verifica che il ruolo IAM associato alla destinazione disponga delle autorizzazioni per Firehose.
Il filtro di sottoscrizione non è riuscito o è bloccato nello stato In sospeso.	Verifica quanto segue: 1. Nell'account Application, verifica che il ruolo IAM abbia: • Autorizzazioni per chiamare`PutSubscriptionFilter`. • Un rapporto di fiducia con CloudWatch Logs. 2. Verificare che l'ARN di destinazione sia corretto. 3. Controlla CloudWatch i log per messaggi di errore specifici.
Lo stream di distribuzione di Firehose non mostra alcun record in entrata.	Verificare quanto segue: 1. Verificare che il ruolo IAM di Firehose abbia: • Autorizzazioni per scrivere su Amazon S3.

Problema	Soluzione
	• Accesso alla AWS KMS chiave se la crittografia è abilitata. 2. Rivedi le CloudWatch metriche per: • `IncomingRecords` • `DeliveryToS3.Records` 3. Verifica le impostazioni del buffer e le configurazioni di consegna.

Risorse correlate

- Configurazione dell'infrastruttura Terraform (documentazione Terraform)
- Implementare AWS Control Tower Account Factory for Terraform (AFT) (AWS Control Tower documentazione)
- Tutorial IAM: delega l'accesso Account AWS utilizzando i ruoli IAM () IAMdocumentation

Configura end-to-end la crittografia per le applicazioni su Amazon EKS utilizzando cert-manager e Let's Encrypt

Mahendra Revanasiddappa e Vasanth Jeyaraj, Amazon Web Services

Riepilogo

L'implementazione della end-to-end crittografia può essere complessa e devi gestire i certificati per ogni risorsa nell'architettura dei microservizi. Sebbene sia possibile interrompere la connessione Transport Layer Security (TLS) ai margini della rete Amazon Web Services (AWS) con un Network Load Balancer o Amazon API Gateway, alcune organizzazioni richiedono la crittografia. end-to-end

Questo modello utilizza NGINX Ingress Controller per l'ingresso. Questo perché quando crei un ingresso Kubernetes, la risorsa di ingresso utilizza un Network Load Balancer. Il Network Load Balancer non consente il caricamento di certificati client. Pertanto, non puoi ottenere un TLS reciproco con Kubernetes Ingress.

Questo modello è destinato alle organizzazioni che richiedono l'autenticazione reciproca tra tutti i microservizi delle loro applicazioni. Mutual TLS riduce l'onere di mantenere i nomi utente o le password e può anche utilizzare il framework di sicurezza chiavi in mano. L'approccio di questo modello è compatibile se l'organizzazione ha un gran numero di dispositivi connessi o deve rispettare rigide linee guida di sicurezza.

Questo modello aiuta a migliorare il livello di sicurezza dell'organizzazione implementando la end-to-end crittografia per le applicazioni in esecuzione su Amazon Elastic Kubernetes Service (Amazon EKS). Questo modello fornisce un'applicazione e un codice di esempio nel repository di GitHub End-to-end crittografia su Amazon EKS per mostrare come un microservizio funziona con la end-to-end crittografia su Amazon EKS. L'approccio del pattern utilizza cert-manager, un componente aggiuntivo di Kubernetes, con Let's Encrypt come autorità di certificazione (CA). Let's Encrypt è una soluzione economica per gestire i certificati e fornisce certificati gratuiti validi per 90 giorni. Cert-manager automatizza il provisioning e la rotazione su richiesta dei certificati quando viene distribuito un nuovo microservizio su Amazon EKS.

Destinatari

Questo modello è consigliato agli utenti che hanno esperienza con Kubernetes, TLS, Amazon Route 53 e Domain Name System (DNS).

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un cluster Amazon EKS esistente.
- AWS Command Line Interface (AWS CLI) versione 1.7 o successiva, installata e configurata su macOS, Linux o Windows.
- L'utilità da riga di `kubectl` comando, installata e configurata per accedere al cluster Amazon EKS. Per ulteriori informazioni su questo argomento, consulta Installazione di kubectl nella documentazione di Amazon EKS.
- Un nome DNS esistente per testare l'applicazione. Per ulteriori informazioni su questo argomento, consulta Registrazione di nomi di dominio utilizzando Amazon Route 53 nella documentazione di Amazon Route 53.
- L'ultima versione di Helm, installata sul tuo computer locale. Per ulteriori informazioni su questo argomento, consulta Using Helm with Amazon EKS nella documentazione di Amazon EKS e nel repository GitHub Helm.
- La GitHub End-to-end crittografia nel repository Amazon EKS, clonata sul tuo computer locale.
- Sostituisci i seguenti valori nei `trustpolicy.json` file `policy.json` and dalla GitHub End-to-end crittografia clonata sull'archivio Amazon EKS:
  - `<account number>`— Sostituiscilo con l'ID dell'account AWS per l'account in cui desideri implementare la soluzione.
  - `<zone id>`— Sostituire con l'ID di zona Route 53 del nome di dominio.
  - `<node_group_role>`— Sostituire con il nome del ruolo AWS Identity and Access Management (IAM) associato ai nodi Amazon EKS.
  - `<namespace>`— Sostituiscilo con lo spazio dei nomi Kubernetes in cui distribuisci il controller di ingresso NGINX e l'applicazione di esempio.
  - `<application-domain-name>`— Sostituire con il nome di dominio DNS di Route 53.

Limitazioni

- Questo modello non descrive come ruotare i certificati e dimostra solo come utilizzare i certificati con microservizi su Amazon EKS.

Architecture

Il diagramma seguente mostra i componenti del flusso di lavoro e dell'architettura di questo modello.

Il diagramma mostra il flusso di lavoro seguente:

1. Un client invia una richiesta di accesso all'applicazione al nome DNS.
2. Il record Route 53 è un CNAME per il Network Load Balancer.
3. Il Network Load Balancer inoltra la richiesta al controller di ingresso NGINX configurato con un listener TLS. La comunicazione tra NGINX Ingress Controller e Network Load Balancer segue il protocollo HTTPS.
4. Il NGINX Ingress Controller esegue il routing basato sul percorso in base alla richiesta del client al servizio applicativo.
5. Il servizio applicativo inoltra la richiesta al pod dell'applicazione. L'applicazione è progettata per utilizzare lo stesso certificato chiamando segreti.
6. I pod eseguono l'applicazione di esempio utilizzando i certificati cert-manager. La comunicazione tra NGINX Ingress Controller e i pod utilizza HTTPS.

Nota: Cert-Manager viene eseguito nel proprio spazio dei nomi. Utilizza un ruolo del cluster Kubernetes per fornire certificati come segreti in namespace specifici. Puoi collegare questi namespace ai pod delle applicazioni e al NGINX Ingress Controller.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Kubernetes Service (Amazon EKS) è un servizio gestito che puoi usare per eseguire Kubernetes su AWS senza dover installare, gestire e mantenere il tuo piano di controllo o i tuoi nodi Kubernetes.
- Elastic Load Balancing distribuisce automaticamente il traffico in entrata su più destinazioni, contenitori e indirizzi IP.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon Route 53 è un servizio Web DNS altamente scalabile e disponibile.

Altri strumenti

- cert-manager è un componente aggiuntivo di Kubernetes che richiede certificati, li distribuisce nei contenitori Kubernetes e automatizza il rinnovo dei certificati.
- NGINX Ingress Controller è una soluzione di gestione del traffico per app native del cloud in Kubernetes e ambienti containerizzati.

Epiche

Crea e configura una zona ospitata pubblica con Route 53

Operazione	Description	Competenze richieste
Crea una zona ospitata pubblica in Route 53.	Accedi alla Console di gestione AWS, apri la console Amazon Route 53, scegli Zone ospitate, quindi scegli Crea zona ospitata. Crea una zona ospitata pubblica e registra l'ID della zona. Per ulteriori informazioni su questo argomento, consulta Creazione di una zona ospitata pubblica nella documentazione di Amazon Route 53. **Note** ACME DNS01 utilizza il provider DNS per inviare una richiesta di rilascio del certifica to al cert-manager. Questa sfida ti chiede di dimostrare di controllare il DNS del	AWS DevOps

Operazione	Description	Competenze richieste
	tuo nome di dominio inserendo un valore specifico in un record TXT sotto quel nome di dominio. Dopo che Let's Encrypt ha assegnato un token al tuo client ACME, quest'ultimo crea un record TXT derivato da quel token e dalla chiave del tuo account, e inserisce quel record in. `_acme-challenge.<YOURDOMAIN>` Quindi Let's Encrypt interroga il DNS per quel record. Se trova una corrispondenza, puoi procedere all'emissione di un certificato.	

Configura un ruolo IAM per consentire al cert-manager di accedere alla zona pubblica ospitata

Operazione	Description	Competenze richieste
Crea la policy IAM per cert-manager.	È necessaria una policy IAM per fornire al cert-manager l'autorizzazione a convalidare la proprietà del dominio	AWS DevOps

Operazione	Description	Competenze richieste
	Route 53. La policy IAM di `policy.json` esempio è fornita nella `1-IAMRole` directory del repository di GitHub End-to-end crittografia clonata su Amazon EKS. Inserisci il seguente comando nella CLI di AWS per creare la policy IAM. `aws iam create-policy \ --policy-name PolicyForCertManager \ --policy-document file://policy.json`	
Crea il ruolo IAM per cert-manager.	Dopo aver creato la policy IAM, devi creare un ruolo IAM. Il ruolo IAM di `trustpolicy.json` esempio è fornito nella `1-IAMRole` directory. Inserisci il seguente comando nella CLI di AWS per creare il ruolo IAM. `aws iam create-role \ --role-name RoleForCertManager \ --assume-role-policy-document file://trustpolicy.json`	AWS DevOps

Operazione	Description	Competenze richieste
Collegare la policy al ruolo.	Inserisci il seguente comando nella CLI di AWS per collegare la policy IAM al ruolo IAM. Sostituiscilo `AWS_ACCOUNT_ID` con l'ID del tuo account AWS. `aws iam attach-role-policy \ --policy-arn arn:aws:iam::AWS_ACCOUNT_ID:policy/PolicyForCertManager \ --role-name RoleForCertManager`	AWS DevOps

Configurazione del controller di ingresso NGINX in Amazon EKS

Operazione	Description	Competenze richieste
Implementa il controller di ingresso NGINX.	Installa la versione più recente di utilizzo di Helm. `nginx-ingress` È possibile modificare la `nginx-ingress` configurazione in base alle proprie esigenze prima di distribuirla. Questo modello utilizza un Network Load Balancer annotato e rivolto internamente, disponibile nella directory. `5-Nginx-Ingress-Controller` Installa il controller di ingresso NGINX eseguendo il seguente	AWS DevOps

Operazione	Description	Competenze richieste
	comando Helm dalla directory `. 5-Nginx-Ingress-Controller` `helm install test-nginx nginx-stable/nginx-ingress -f 5-Nginx-Ingress-Controller/values_internal_nlb.yaml`	
Verifica che il controller di ingresso NGINX sia installato.	Immettere il comando `helm list`. L'output dovrebbe mostrare che il NGINX Ingress Controller è installato.	AWS DevOps

Operazione	Description	Competenze richieste
Crea un record Route 53 A.	Il record A punta al Network Load Balancer creato da NGINX Ingress Controller. 1. Ottieni il nome DNS del Network Load Balancer. Per istruzioni, consulta Ottenere il nome DNS per un sistema di bilanciamento del carico ELB. 2. Sulla console Amazon Route 53, scegli Hosted Zones. 3. Seleziona la zona ospitata pubblica in cui desideri creare il record, quindi scegli Crea record. 4. Inserisci un nome per il record. 5. In Tipo di record, scegli A - Indirizza il traffico verso IPv4 e alcune risorse AWS. 6. Abilita Alias. 7. In Indirizza il traffico verso, procedi come segue: a. Scegli Alias to Network Load Balancer. b. Scegli la regione AWS in cui viene distribuito il Network Load Balancer.	AWS DevOps

Operazione	Description	Competenze richieste
	c. Immettere il nome DNS del Network Load Balancer. 8. Scegli Crea record.	

Configurare NGINX VirtualServer su Amazon EKS

Operazione	Description	Competenze richieste
Implementa NGINX VirtualServer.	La VirtualServer risorsa NGINX è una configurazione di bilanciamento del carico alternativa alla risorsa in ingresso. La configurazione per creare la VirtualServer risorsa NGINX è disponibile nel file nella directory. `nginx_virtualserver.yaml 6-Nginx-Virtual-Server` Immettete il seguente comando `kubectl` per creare la risorsa VirtualServer NGINX. `kubectl apply -f nginx_virtualserver.yaml` ⚠ Important Assicurati di aggiornare il nome di dominio dell'applicazione, il	AWS DevOps

Operazione	Description	Competenze richieste
	segreto del certificato e il nome del servizio dell'applicazione nel `nginx_virtualserver.yaml` file.	
Verifica che NGINX VirtualServer sia stato creato.	Inserisci il seguente comando `kubectl` per verificare che la VirtualServer risorsa NGINX sia stata creata correttamente. `kubectl get virtualserver` **Note** Verifica che la `Host` colonna corrisponda al nome di dominio dell'applicazione.	AWS DevOps

Operazione	Description	Competenze richieste
Implementa il server web NGINX con TLS abilitato.	Questo modello utilizza un server web NGINX con TLS abilitato come applicazione per testare la crittografia. end-to-end I file di configura zione necessari per distribui re l'applicazione di test sono disponibili nella directory. `demo-webserver` Immettete il seguente comando `kubectl` per distribuire l'applicazione di test. <br`kubectl apply -f nginx-tls-ap.yaml`	AWS DevOps

Operazione	Description	Competenze richieste
Verifica che le risorse dell'applicazione di test siano state create.	Immettete i seguenti comandi `kubectl` per verificare che vengano create le risorse richieste per l'applicazione di test: • `kubectl get deployments` **Note** Convalida la `Ready` colonna e la `Available` colonna. • `kubectl get pods | grep -i example-deploy` **Note** I pod devono essere in `running` stato. • `kubectl get configmap` • `kubectl get svc`	AWS DevOps

Operazione	Description	Competenze richieste
Convalida l'applicazione.	1. Immettete il seguente comando sostituendolo `<application-domain-name>` con il nome DNS Route53 creato in precedenza. `curl --verbose https://<application-domain-name>` 2. Verifica di poter accedere all'applicazione.	AWS DevOps

Risorse correlate

Risorse AWS

- Creazione di record utilizzando la console Amazon Route 53 (documentazione Amazon Route 53)
- Utilizzo di un Network Load Balancer con il controller di ingresso NGINX su Amazon EKS (post sul blog AWS)

Altre risorse

- Route 53 (documentazione del cert-manager)
- Configurazione di DNS01 Challenge Provider (documentazione cert-manager)
- Sfida Let's encrypt DNS (documentazione Let's Encrypt)

Semplifica la distribuzione di applicazioni multi-tenant Amazon EKS utilizzando Flux

Nadeem Rahaman, Aditya Ambati, Aniket Dekate e Shrikant Patil, Amazon Web Services

Riepilogo

Molte aziende che offrono prodotti e servizi sono settori regolamentati dai dati, tenuti a mantenere le barriere relative ai dati tra le loro funzioni aziendali interne. Questo modello descrive come utilizzare la funzionalità multi-tenancy di Amazon Elastic Kubernetes Service (Amazon EKS) per creare una piattaforma dati che consenta l'isolamento logico e fisico tra tenant o utenti che condividono un singolo cluster Amazon EKS. Il modello fornisce l'isolamento attraverso i seguenti approcci:

- Isolamento dello spazio dei nomi Kubernetes
- Controllo degli accessi basato sui ruoli (RBAC)
- Policy di rete
- Quote delle risorse
- AWS Identity and Access Management ruoli (IAM) per gli account di servizio (IRSA)

Inoltre, questa soluzione utilizza Flux per mantenere immutabile la configurazione del tenant durante la distribuzione delle applicazioni. È possibile distribuire le applicazioni tenant specificando il repository tenant che contiene il file Flux nella configurazione. `kustomization.yaml`

Questo modello implementa quanto segue:

- Un AWS CodeCommit repository, AWS CodeBuild progetti e una AWS CodePipeline pipeline, creati distribuendo manualmente gli script Terraform.
- Componenti di rete e di calcolo necessari per ospitare i tenant. Questi vengono creati tramite CodePipeline e CodeBuild utilizzando Terraform.
- Namespace dei tenant, politiche di rete e quote di risorse, configurati tramite un grafico Helm.
- Applicazioni che appartengono a tenant diversi, distribuite utilizzando Flux.

Ti consigliamo di pianificare e creare attentamente la tua architettura per la multi-tenancy in base ai tuoi requisiti unici e alle tue considerazioni di sicurezza. Questo modello fornisce un punto di partenza per l'implementazione.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- AWS Command Line Interface (AWS CLI) versione 2.11.4 o successiva, installata e configurata
- Terraform versione 0.12 o successiva installata sul computer locale
- Terraform AWS Provider versione 3.0.0 o successiva
- Kubernetes Provider versione 2.10 o successiva
- Helm Provider versione 2.8.0 o successiva
- Kubectl Provider versione 1.14 o successiva

Limitazioni

- Dipendenza dalle distribuzioni manuali di Terraform: la configurazione iniziale del flusso di lavoro, inclusa la creazione di CodeCommit repository, CodeBuild progetti e pipeline, si basa sulle implementazioni manuali di Terraform. CodePipeline Ciò introduce una potenziale limitazione in termini di automazione e scalabilità, poiché richiede un intervento manuale per le modifiche all'infrastruttura.
- CodeCommit dipendenza dal repository: il flusso di lavoro si basa sui CodeCommit repository come soluzione di gestione del codice sorgente ed è strettamente associato a. Servizi AWS

Architecture

Architetture di destinazione

Questo modello implementa tre moduli per creare la pipeline, la rete e l'infrastruttura di calcolo per una piattaforma dati, come illustrato nei diagrammi seguenti.

Architettura della pipeline:

Architettura di rete:

Architettura di calcolo:

Tools (Strumenti)

Servizi AWS

- AWS CodeBuildè un servizio di compilazione completamente gestito che ti aiuta a compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS CodeCommitè un servizio di controllo delle versioni che consente di archiviare e gestire in modo privato gli archivi Git, senza dover gestire il proprio sistema di controllo del codice sorgente.
- AWS CodePipelineti aiuta a modellare e configurare rapidamente le diverse fasi di una versione del software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire Kubernetes su AWS Kubernetes senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- AWS Transit Gatewayè un hub centrale che collega cloud privati virtuali () e reti locali. VPCs
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.

Altri strumenti

- Le politiche di rete Cilium supportano le politiche di rete Kubernetes L3 e L4. Possono essere estesi con policy L7 per fornire sicurezza a livello di API per HTTP, Kafka e gRPC e altri protocolli simili.
- Flux è uno strumento di distribuzione continua (CD) basato su Git che automatizza le implementazioni delle applicazioni su Kubernetes.
- Helm è un gestore di pacchetti open source per Kubernetes che ti aiuta a installare e gestire le applicazioni sul tuo cluster Kubernetes.
- Terraform è uno strumento di infrastruttura come codice (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.

Archivio di codici

Il codice per questo pattern è disponibile nel repository GitHub EKS Multi-Tenancy Terraform Solution.

Best practice

Per le linee guida e le migliori pratiche per l'utilizzo di questa implementazione, consulta quanto segue:

- Best practice per la multi-tenancy di Amazon EKS
- Documentazione Flux

Epiche

Crea pipeline per le fasi di costruzione, test e distribuzione di Terraform

Operazione	Description	Competenze richieste
Clona il repository del progetto.	Clona il repository GitHub EKS Multi-Tenancy Terraform Solution eseguendo il seguente comando in una finestra di terminale: `git clone https://github.com/aws-samples/aws-eks-multitenancy-deployment.git`	AWS DevOps
Avvia il bucket Terraform S3 e Amazon DynamoDB.	1. Nella `bootstrap` cartella, apri il `bootstrap.sh` file e aggiorna i valori delle variabili per il nome del bucket S3, il nome della tabella DynamoDB e: Regione AWS `S3_BUCKET_NAME="<S3_BUCKET_NAME>" DYNAMODB_TABLE_NAME="<DYNAMODB_NAME>"`	AWS DevOps

Operazione	Description	Competenze richieste

	`REGION="<AWS_REGION>"`
	2. Eseguire lo script `bootstrap.sh` . Lo script richiede il AWS CLI, che hai installato come parte dei prerequisiti.
	`cd bootstrap`
	`./bootstrap.sh`	

Operazione	Description	Competenze richieste

Aggiorna i `locals.tf` file `run.sh` and.	1. Una volta completato correttamente il processo di bootstrap, copia il bucket S3 e il nome della tabella DynamoDB dalla sezione dello script: `variables bootstrap.sh`
	`# Variables S3_BUCKET_NAME="< S3_BUCKET_NAME>" DYNAMODB_TABLE_NAME ="<DYNAMODB_NAME"`
	2. Incolla questi valori nello `run.sh` script, che si trova nella directory principale del progetto:
	`BACKEND_BUCKET_ID= "<SAME_NAME_AS_S3_ BUCKET_NAME>" DYNAMODB_ID=" <SAME_NAME_AS_DYNA MODB_NAME>"`
	3. Carica il codice del progetto in un CodeCommit repositor y. Puoi creare automatic amente questo repositor y tramite Terraform impostando la seguente variabile su `true` nel file: `demo/pipeline/loca ls.tf`
		AWS DevOps

Operazione	Description	Competenze richieste
	`create_new_repo = true` 4. Aggiorna il `locals.tf` file in base alle tue esigenze per creare risorse di pipeline.	

Implementa il modulo pipeline.	Per creare risorse di pipeline, esegui manualmente i seguenti comandi Terraform . Non esiste alcuna orchestrazione per l'esecuzione automatica di questi comandi.
	`./run.sh -m pipeline -e demo -r <AWS_REGION> -t init`
	`./run.sh -m pipeline -e demo -r <AWS_REGION> -t plan`
	`./run.sh -m pipeline -e demo -r <AWS_REGION> -t apply`
		AWS DevOps

Crea l'infrastruttura di rete

Operazione	Description	Competenze richieste
Avvia la pipeline.	1. Nella `templates` cartella, assicuratevi che per i `buildspec` file sia impostata la seguente variabile su: `network`	AWS DevOps

Operazione	Description	Competenze richieste
	`TF_MODULE_TO_BUILD: "network"` 2. Sulla CodePipeline console, nella pagina dei dettagli della pipeline, avvia la pipeline scegliendo Release change. Dopo questa prima esecuzione, la pipeline si avvia automaticamente ogni volta che esegui una modifica al ramo principale del CodeCommit repository. La pipeline include le seguenti fasi: • `validate`inizializza Terraform, esegue le scansioni di sicurezza Terraform utilizzando gli strumenti checkov e tfsec e carica i report di scansione nel bucket S3. • `plan` mostra il piano Terraform e carica il piano nel bucket S3. • `apply`applica l'output del piano Terraform dal bucket S3 e crea risorse. AWS • `destroy`rimuove le AWS risorse create durante la	

Operazione	Description	Competenze richieste
	fase. `apply` Per abilitare questa fase opzionale , imposta la seguente variabile su `true` nel `demo/pipeline/locals.tf` file: `enable_destroy_stage = true`	

Operazione	Description	Competenze richieste
Convalida le risorse create tramite il modulo di rete.	Verifica che le seguenti AWS risorse siano state create dopo la corretta implementazione della pipeline: • Un VPC in uscita con tre sottoreti pubbliche e tre private, gateway Internet e gateway NAT. • Un VPC Amazon EKS con tre sottoreti private. • Tenant 1 e Tenant 2 VPCs con tre sottoreti private ciascuna. • Un gateway di transito con tutti gli allegati VPC e i percorsi verso ogni sottorete privata. • Un gateway di transito statico per il VPC di uscita Amazon EKS con un blocco CIDR di destinazione di. `0.0.0.0/0` Ciò è necessario per consentire a tutti di accedere VPCs a Internet in uscita tramite il VPC di uscita di Amazon EKS.	AWS DevOps

Crea l'infrastruttura di elaborazione

Operazione	Description	Competenze richieste

Aggiorna `locals.tf` per consentire l'accesso del CodeBuild progetto al VPC.	Per distribuire i componenti aggiuntivi per il cluster privato Amazon EKS, il CodeBuild progetto deve essere collegato al VPC Amazon EKS.
	1. Nella `demo/pipeline` cartella, apri il `locals.tf` file e imposta la `vpc_enabled` variabile su. `true`
	2. Esegui lo `run.sh` script per applicare le modifiche al modulo pipeline:
	`demo/pipeline/locals.tf`
	`./run.sh -m pipeline -env demo -region <AWS_REGION> -tfcmd init`
	`./run.sh -m pipeline -env demo -region <AWS_REGION> -tfcmd plan`
	`./run.sh -m pipeline -env demo -region <AWS_REGION> -tfcmd apply`
		AWS DevOps

Aggiorna i `buildspec` file per creare il modulo di calcolo.	Nella `templates` cartella, in tutti i file `buildspec` YAML, imposta il valore della `TF_MODULE_TO_BUILD`	AWS DevOps

Operazione	Description	Competenze richieste
	variabile da a: `network` `compute` `TF_MODULE_TO_BUILD: "compute"`	

Operazione	Description	Competenze richieste

Aggiorna il `values` file per il diagramma Helm di gestione dei tenant.	1. Aprire il `values.yaml` file nella seguente posizione:
	`cd cfg-terraform/demo/compute/cfg-tenant-mgmt`
	Il file ha il seguente aspetto:
	`---`
	`global:`
	`  clusterRoles:`
	`  operator: platform-tenant`
	`  flux: flux-tenant-applier`
	`  flux:`
	`  tenantCloneBaseUrl: ${TEANT_BASE_URL}`
	`  repoSecret: ${TENANT_REPO_SECRET}`
	`tenants:`
	`  tenant-1:`
	`  quotas:`
	`  limits:`
	`  cpu: 1`
	`  memory: 1Gi`
	`  flux:`
	`  path: overlays/tenant-1`
	`  tenant-2:`
	`  quotas:`
	`  limits:`
	`  cpu: 1`
	`  memory: 2Gi`
	`  flux:`
		AWS DevOps

Operazione	Description	Competenze richieste

	<pre>path: overlays/
	tenant-2</pre>
	2. Nelle `tenants` sezioni `global` e, aggiorna la configurazione in base alle tue esigenze:
	• `tenantCloneBaseUrl` — Percorso del repository che ospita il codice per tutti i tenant (utilizziamo lo stesso repository Git per tutti i tenant)
	• `repoSecret` — Segreto Kubernetes che contiene le chiavi SSH e gli host noti per l'autenticazione nel repository Git dei tenant globali
	• `quotas`— Quote di risorse Kubernetes da applicare a ciascun tenant
	• `flux path`— Percorso dei file YAML dell'appl icazione tenant nel repository globale dei tenant	

Operazione	Description	Competenze richieste
Convalida le risorse di calcolo.	Dopo aver aggiornato i file nei passaggi precedenti, si CodePipeline avvia automatic amente. Verifica che abbia creato le seguenti AWS risorse per l'infrastruttura di elaborazione: • Cluster Amazon EKS con endpoint privato • Nodi di lavoro Amazon EKS • Componenti aggiuntiv i Amazon EKS: segreti esterni `aws-loadb alancer-controller` e `metrics-server` • GitOps modulo, diagramma Flux Helm, diagramma Cilium Helm e tabella Helm per la gestione degli inquilini	AWS DevOps

Controlla la gestione dei tenant e altre risorse

Operazione	Description	Competenze richieste
Convalida le risorse di gestione dei tenant in Kubernetes.	Esegui i seguenti comandi per verificare che le risorse di gestione dei tenant siano state create correttamente con l'aiuto di Helm. 1. I namespace dei tenant sono stati creati, come	AWS DevOps

Operazione	Description	Competenze richieste

	specificato in: `values.yaml`
	
	`kubectl get ns -A`
	
	2. Le quote vengono assegnate a ogni spazio dei nomi dei tenant, come specificato in: `values.yaml`
	
	`kubectl get quota --namespace=<tenant_namespace>`
	
	3. I dettagli delle quote sono corretti per ogni spazio dei nomi dei tenant:
	
	`kubectl describe quota cpu-memory-resource-quota-limit -n <tenant_namespace>`
	
	4. Le politiche di rete Cilium sono state applicate a ogni spazio dei nomi dei tenant:
	
	`kubectl get CiliumNetworkPolicy -A`	

Operazione	Description	Competenze richieste

Verifica le distribuzioni delle applicazioni tenant.	Esegui i seguenti comandi per verificare che le applicazioni tenant siano state distribuite.
	
	1. Flux è in grado di connettersi al CodeCommit repository specificato nel modulo: GitOps
	
	`kubectl get gitrepositories -A`
	
	2. Il controller di personalizzazione Flux ha distribuito i file YAML nel repository: CodeCommit
	
	`kubectl get kustomizations -A`
	
	3. Tutte le risorse dell'applicazione vengono distribuite nei relativi namespace dei tenant:
	
	`kubectl get all -n <tenant_namespace>`
	
	4. È stato creato un ingresso per ogni tenant:
	
	`kubectl get ingress -n <tenant_namespace>`	

risoluzione dei problemi

Problema	Soluzione

Viene visualizzato un messaggio di errore simile al seguente:
 	
`Failed to checkout and determine revision: unable to clone unknown error: You have successfully authenticated over SSH. You can use Git to interact with AWS CodeCommit.`
	Segui questi passaggi per risolvere il problema:
	
	1. Verifica l'archivio delle applicazioni tenant: un repository vuoto o configurato in modo errato potrebbe causare l'errore. Assicurati che il repository dell'applicazione tenant contenga il codice richiesto.
	2. Ridistribuisci il `tenant_mgmt` modulo: nel file di configurazione del `tenant_mgmt` modulo, individua il `app` blocco, quindi imposta il parametro su: `deploy 0`
	
	`deploy = 0`
	
	Dopo aver eseguito il `apply` comando Terraform, modifica nuovamente il valore del `deploy` parametro in: `1`
	
	`deploy = 1`
	
	3. Ricontrolla lo stato: dopo aver eseguito i passaggi precedenti, utilizza il seguente comando per verificare se il problema persiste:
	
	`kubectl get gitrepositories -A`
	
	Se il problema persiste, valuta la possibilità di approfondire i log di Flux per maggiori dettagli o consulta la guida generale alla risoluzione dei problemi di Flux.

Risorse correlate

- Progetti Amazon EKS per Terraform
- Guide alle best practice di Amazon EKS, sezione Multi-tenancy
- Sito web Flux
- Sito web Helm

Informazioni aggiuntive

Ecco un esempio di struttura di repository per la distribuzione di applicazioni tenant:

```
applications
sample_tenant_app
### README.md
### base
#   ### configmap.yaml
#   ### deployment.yaml
#   ### ingress.yaml
#   ### kustomization.yaml
#   ### service.yaml
### overlays
    ### tenant-1
    #   ### configmap.yaml
    #   ### deployment.yaml
    #   ### kustomization.yaml
    ### tenant-2
        ### configmap.yaml
        ### kustomization.yaml
```

Semplifica lo sviluppo e la distribuzione di bot Amazon Lex utilizzando un flusso di lavoro automatizzato

Balaji Panneerselvam, Attila Dancso, Pavan Dusanapudi, Anand Jumnani e James O'Hara, Amazon Web Services

Riepilogo

Lo sviluppo e la distribuzione di bot conversazionali Amazon Lex possono essere difficili quando si cerca di gestire più funzionalità, sviluppatori e ambienti. Un flusso di lavoro automatizzato che utilizza i principi dell'infrastruttura come codice (IaC) può aiutare a semplificare il processo. Questo modello può aiutare a migliorare la produttività degli sviluppatori di Amazon Lex e consentire una gestione efficiente del ciclo di vita dei bot nei seguenti modi:

- Abilita lo sviluppo simultaneo di più funzionalità - Con un flusso di lavoro automatizzato, gli sviluppatori possono lavorare su diverse funzionalità in parallelo in rami separati. Le modifiche possono quindi essere unite e implementate senza bloccare altre attività.
- Usa l'interfaccia utente della console Amazon Lex: gli sviluppatori possono utilizzare l'intuitiva console Amazon Lex per creare e testare bot. I bot vengono quindi descritti nel codice di infrastruttura per la distribuzione.
- Promuovi i bot in tutti gli ambienti: il flusso di lavoro automatizza la promozione delle versioni dei bot da ambienti inferiori, come quelli di sviluppo e test, fino alla produzione. Questo approccio riduce il rischio e il sovraccarico delle promozioni manuali.
- Mantieni il controllo delle versioni: la gestione delle definizioni dei bot in Git anziché esclusivamente tramite il servizio Amazon Lex ti offre il controllo della versione e un audit trail. Le modifiche vengono tracciate in base ai singoli sviluppatori, a differenza di quando si utilizzano solo Console di gestione AWS o si modificano APIs i bot memorizzati in. AWS

Automatizzando il processo di rilascio dei bot di Amazon Lex, i team possono fornire funzionalità più rapidamente con rischi e sforzi ridotti. I bot rimangono sotto il controllo della versione anziché isolati nella console Amazon Lex.

Prerequisiti e limitazioni

Prerequisiti

- Il flusso di lavoro Account AWS prevede diversi ambienti (sviluppo, produzione e DevOps), il che richiede la gestione degli account e configurazioni di accesso tra account.

- Python 3.9 disponibile nell'ambiente di distribuzione o nella pipeline.
- Git è installato e configurato su una workstation locale per il controllo del codice sorgente.
- AWS Command Line Interface (AWS CLI) installato e configurato per l'autenticazione utilizzando la riga di comando o Python.

Limitazioni

- Accesso al repository: il flusso di lavoro presuppone che la pipeline di integrazione e distribuzione continua (CI/CD) disponga delle autorizzazioni necessarie per apportare modifiche all'archivio del codice sorgente.
- Versione iniziale del bot: gli strumenti richiedono che venga distribuita una versione iniziale del bot utilizzando modelli. AWS CloudFormation È necessario creare la prima iterazione del bot e salvarla nel repository prima che il flusso di lavoro automatizzato possa prendere il sopravvento.
- Conflitti di fusione: sebbene il flusso di lavoro miri a consentire lo sviluppo simultaneo, esiste ancora la possibilità di unire i conflitti quando si integrano modifiche provenienti da diversi rami. La risoluzione dei conflitti nelle configurazioni dei bot potrebbe richiedere un intervento manuale.

Versioni del prodotto

- Python 3.9 o versioni successive
- AWS CDK v2 2.124.0 o superiore
- AWS SDK per Python (Boto3)1.28 o superiore

Architecture

Il diagramma seguente mostra l'architettura di alto livello e i componenti chiave della soluzione.

I componenti chiave includono quanto segue:

- Lex bot repo: un repository Git che memorizza le definizioni IaC per i bot Amazon Lex.
- DevOps— Account AWS Dedicato all'alloggiamento delle CI/CD pipeline e delle relative risorse per il processo di sviluppo e implementazione.
- Pipeline: le AWS CodePipeline istanze che automatizzano varie fasi del ciclo di vita dello sviluppo e della distribuzione dei bot, come la creazione di un nuovo bot, l'esportazione della definizione di un bot, l'importazione di una definizione di bot e l'eliminazione di un bot.

- Ticket bot e bot principale: le risorse bot di Amazon Lex, in cui i ticket bot sono bot con funzionalità specifiche sviluppati da singoli team o sviluppatori e il bot principale è il bot di base che integra tutte le funzionalità.

Il diagramma dell'architettura illustra il seguente flusso di lavoro:

1. Bot principale di base: il punto di partenza del flusso di lavoro è quello di basare il bot principale nell'ambiente di sviluppo (Dev). Il bot principale funge da base per lo sviluppo futuro e l'aggiunta di funzionalità.
2. Crea ticket bot: quando è richiesta una nuova funzionalità o una modifica, viene creato un ticket bot. Il ticket bot è essenzialmente una copia o un ramo del bot principale su cui gli sviluppatori possono lavorare senza influire sulla versione principale.
3. Export ticket bot: una volta completato il lavoro sul ticket bot, questo viene esportato dal servizio Amazon Lex. Quindi, la filiale che contiene il ticket bot viene sostituita dalla filiale principale. Questo passaggio assicura che tutte le modifiche apportate al bot principale durante lo sviluppo del ticket bot vengano incorporate, riducendo i potenziali conflitti.
4. Importa ticket bot ribasato e convalida: il ticket bot ribasato viene reimportato nell'ambiente di sviluppo e convalidato per garantire che funzioni correttamente con le ultime modifiche apportate dalla filiale principale. Se la convalida ha esito positivo, viene creata una pull request (PR) per unire le modifiche apportate al ticket bot nel ramo principale.
5. Elimina ticket bot: dopo che le modifiche sono state integrate con successo nel ramo principale, il ticket bot non è più necessario. Il ticket bot può essere eliminato per mantenere l'ambiente pulito e gestibile.
6. Implementa il bot principale nell'ambiente di sviluppo e verifica: il bot principale aggiornato, che ora include le nuove funzionalità o modifiche, viene distribuito nell'ambiente di sviluppo. Qui viene sottoposto a test approfonditi per garantire che tutte le funzionalità funzionino come previsto.
7. Implementazione del bot principale nell'ambiente di produzione: una volta completati e completati con successo i test nell'ambiente di sviluppo, il bot principale viene distribuito nell'ambiente di produzione. Questo passaggio è la fase finale del flusso di lavoro, in cui le nuove funzionalità diventano disponibili per gli utenti finali.

Automazione e scalabilità

Il flusso di lavoro automatizzato consente agli sviluppatori di lavorare su diverse funzionalità in parallelo, ciascuna in rami separati. Ciò facilita lo sviluppo simultaneo, consentendo ai team di

collaborare in modo efficace e fornire funzionalità più velocemente. Con le filiali isolate l'una dall'altra, le modifiche possono essere unite e implementate senza bloccare o interferire con altre attività in corso.

Il flusso di lavoro automatizza l'implementazione e la promozione delle versioni dei bot in diversi ambienti, come sviluppo, test e produzione.

L'archiviazione delle definizioni dei bot in un sistema di controllo delle versioni come Git fornisce un audit trail completo e consente una collaborazione efficiente. Le modifiche vengono tracciate per i singoli sviluppatori, garantendo trasparenza e responsabilità durante tutto il ciclo di vita dello sviluppo. Questo approccio facilita anche la revisione del codice, consentendo ai team di identificare e risolvere i problemi prima di passare alla produzione.

Utilizzando AWS CodePipeline e altro Servizi AWS, il flusso di lavoro automatizzato può essere scalato per adattarsi all'aumento dei carichi di lavoro e delle dimensioni dei team.

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software open source per la definizione Cloud AWS dell'infrastruttura in codice utilizzando linguaggi di programmazione familiari e il suo provisioning. CloudFormation L'implementazione di esempio in questo modello utilizza Python.
- AWS CDK Command Line Interface (AWS CDK CLI): il AWS CDK Toolkit è lo strumento principale per interagire con l'app. AWS CDK Esegue l'app, interroga il modello applicativo definito e produce e distribuisce i modelli generati dal CloudFormation CDK.
- AWS CloudFormationti aiuta a configurare AWS le risorse, a fornirle in modo rapido e coerente e a gestirle per tutto il loro ciclo di vita tra e. Account AWS Regioni AWS Questo modello viene utilizzato CloudFormation per distribuire le configurazioni dei bot di Amazon Lex e le relative risorse utilizzando l'infrastruttura come codice.
- AWS CodeBuildè un servizio di compilazione completamente gestito che ti aiuta a compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione. Questo modello viene utilizzato CodeBuild per creare e impacchettare gli artefatti di distribuzione.
- AWS CodePipelineconsente di modellare e configurare rapidamente le diverse fasi di un rilascio del software e di automatizzare i passaggi necessari per rilasciare le modifiche software in modo continuo. Questo modello viene utilizzato CodePipeline per orchestrare la pipeline di distribuzione continua.

- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Lex V2 consente di Servizio AWS creare interfacce conversazionali (bot) per applicazioni che utilizzano voce e testo.
- AWS SDK per Python (Boto3)è un kit di sviluppo software che ti aiuta a integrare la tua applicazione, libreria o script Python con. Servizi AWS

Altri strumenti

- Git è un sistema di controllo delle versioni distribuito open source.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub management-framework-sample-for-amazon-lex. Il repository di codice contiene le seguenti cartelle e file:

- `prerequisite`cartella: contiene le definizioni degli CloudFormation stack (utilizzando il AWS CDK) per configurare le risorse e gli ambienti richiesti.
- `prerequisite/lexmgmtworkflow`folder — Directory principale per il progetto Lex Management Workflow, che include le definizioni dello stack e il codice Python.
- `prerequisite/tests`— Contiene test unitari.
- `src`folder: directory del codice sorgente, che include il wrapper e le utilità di gestione dei bot di Amazon Lex.
- `src/dialogue_lambda`— Directory del codice sorgente della funzione Dialogue Hook Lambda che intercetta ed elabora gli input degli utenti durante una conversazione con un bot Amazon Lex.

Best practice

- Separazione delle preoccupazioni

  - Mantieni una chiara separazione delle responsabilità tra DevOps gli ambienti di sviluppo e produzione.
  - Utilizzali separati Account AWS per ogni ambiente per imporre limiti di isolamento e sicurezza adeguati.
  - Utilizza ruoli tra account diversi e principi di accesso con privilegi minimi per garantire un accesso controllato tra gli ambienti.
- Infrastruttura come codice
  - Rivedi e aggiorna regolarmente il codice dell'infrastruttura per allinearlo alle migliori pratiche e ai requisiti in evoluzione.
  - Stabilisci una chiara strategia di ramificazione e fusione per l'archivio del codice sorgente
- Test e convalida
  - Implementa test automatizzati nelle varie fasi della pipeline per individuare eventuali problemi nelle prime fasi del ciclo di sviluppo.
  - Usa la console Amazon Lex o i framework di test automatizzati per convalidare le configurazioni e le funzionalità dei bot prima di passare ad ambienti superiori.
  - Prendi in considerazione l'implementazione di porte di approvazione manuali per le distribuzioni in ambienti di produzione o critici.
- Monitoraggio e registrazione
  - Imposta meccanismi di monitoraggio e registrazione per le pipeline, le implementazioni e le interazioni con i bot.
  - Monitora gli eventi della pipeline, gli stati di implementazione e le metriche delle prestazioni dei bot per identificare e risolvere tempestivamente i problemi.
  - Utilizza servizi AWS come Amazon CloudWatch e AWS X-Ray per la registrazione e il monitoraggio centralizzati. AWS CloudTrail
  - Esamina e analizza regolarmente le prestazioni, l'efficienza e l'efficacia del flusso di lavoro automatizzato.
- Sicurezza e conformità
  - Implementa pratiche di codifica sicure e segui le best practice AWS di sicurezza per lo sviluppo e la distribuzione di bot Amazon Lex.
  - Rivedi e aggiorna regolarmente i ruoli, le politiche e le autorizzazioni IAM per allinearli al principio del privilegio minimo.
  - Prendi in considerazione l'integrazione delle scansioni di sicurezza e dei controlli di conformità nelle pipeline.

Epiche

Configurare IAC per la gestione dei bot di Amazon Lex

Operazione	Description	Competenze richieste

Configura l'ambiente CDK locale.	1. Per clonare il repository di questo pattern e accedere alla `prerequisite` directory, esegui il seguente comando:
	`git clone https://github.com/aws-samples/management-framework-sample-for-amazon-lex.git`
	`cd management-framework-sample-for-amazon-lex`
	2. Per installare e attivare l'ambiente virtuale Python, esegui il seguente comando che installa le dipendenze del CDK localmente nella cartella del progetto, anziché globalmente:
	`pip install virtualenv`
	`python<version> -m venv .venv`
	`source .venv/bin/activate`
	`python -m pip install -r requirements.txt`
		AWS DevOps

Operazione	Description	Competenze richieste

Crea un ruolo tra account diversi nell'`devops`ambiente.	L'`devops`account è responsabile dell'hosting e della gestione delle CI/CD pipeline. Per consentire alle CI/CD pipeline di interagire con gli `prod` ambienti `dev` and, esegui i seguenti comandi per creare un ruolo tra account nell'account. `devops`
	`cdk bootstrap --profile =devops`
	
	`cdk deploy LexMgmtDevopsRoleStack -c dev-account-id=222222222222 -c prod-account-id=333333333333 --profile=devops`
		AWS DevOps

Operazione	Description	Competenze richieste

Crea un ruolo tra account diversi nell'devambiente.	Crea un ruolo IAM nell'devaccount con le autorizzazioni necessarie per consentire all'devopsaccount di assumere questo ruolo. La CI/CD pipeline utilizza questo ruolo per eseguire azioni nell'devaccount, come la distribuzione e la gestione delle risorse bot di Amazon Lex.
	
	Per creare il ruolo IAM, esegui i seguenti comandi:
	
	`cdk bootstrap --profile =dev`
	
	`cdk deploy LexMgmtCr ossaccountRoleStac k -c devops-account- id=1111111111111 -- profile=dev`
		AWS DevOps

Operazione	Description	Competenze richieste

Crea un ruolo tra account diversi nell'`prod`ambiente.	Crea un ruolo IAM nell'`prod`account con le autorizzazioni necessarie per consentire all'`devops`account di assumere questo ruolo. La CI/CD pipeline utilizza questo ruolo per eseguire azioni nell'`prod`account, come la distribuzione e la gestione delle risorse bot di Amazon Lex.
	`cdk bootstrap --profile =prod`
	
	`cdk deploy LexMgmtCr ossaccountRoleStac k -c devops-account- id=1111111111111 -- profile=prod`
		AWS DevOps

Crea pipeline nell'`devops`ambiente.	Per gestire il flusso di lavoro di sviluppo per i bot Amazon Lex, esegui il comando seguente per configurare le pipeline nell'`devops`ambiente. `cdk deploy LexMgmtWo rkflowStack -c devops- account-id=111111111 1111 -c dev-account- id=2222222222222 -c prod-account-id=33 3333333333 --profile =devops`	AWS DevOps

Stabilisci la linea di base per il bot principale

Operazione	Description	Competenze richieste
Definire la versione iniziale del bot principale.	Per definire la versione iniziale del bot principale, attiva la `BaselineBotPipeline` pipeline. La pipeline implementa la definizione di bot di base definita nel CloudForm ation modello, esporta la definizione del bot principal e come file.json. e archivia il codice del bot principale in un sistema di controllo della versione.	AWS DevOps

Implementa il flusso di lavoro per lo sviluppo

Operazione	Description	Competenze richieste
Crea il ticket bot per sviluppar e e testare una funzionalità.	`TicketBot` è una nuova istanza di bot che viene importata dalla definizione di bot principale esistente nel ramo delle funzionalità. Questo approccio garantisc e che il nuovo bot disponga di tutte le funzionalità e le configurazioni correnti del bot principale. Per definire la versione iniziale del ticket bot, attiva	Sviluppatore Lex Bot

Operazione	Description	Competenze richieste
	la `CreateTicketBotPipeline` pipeline. La pipeline crea un nuovo ramo di funzionalità nel sistema di controllo delle versioni e crea una nuova istanza di ticket bot basata sul bot principale.	
Sviluppa e testa la funzionalità ticket bot.	Per sviluppare e testare la funzionalità, accedi Console di gestione AWS e apri la console Amazon Lex all'indirizzo https://console.aws.amazon.com/lex/. Per ulteriori informazioni, consulta Testare un bot utilizzando la console nella documentazione di Amazon Lex. Con l'`TicketBot` istanza, ora puoi aggiungere, modificare o estendere la funzionalità del bot per implementare la nuova funzionalità. Ad esempio, potete creare o modificare intenti, enunciati, slot e flussi di dialogo. Per ulteriori informazioni, consulta Aggiungere intenti nella documentazione di Amazon Lex.	Sviluppatore Lex Bot

Operazione	Description	Competenze richieste
Esporta la definizione del ticket bot.	La definizione del bot esportata è essenzialmente una rappresentazione della configurazione e della funzionalità del bot in formato JSON. Per esportare la definizione del ticket bot, attiva la `ExportTicketBotPipeline` pipeline. La pipeline esporta la definizione del ticket bot come file.json e memorizza il codice del ticket bot in un ramo di funzionalità del sistema di controllo della versione.	Sviluppatore Lex Bot

Operazione	Description	Competenze richieste
Modifica il ramo delle funzionalità dal ramo principale più recente.	Durante lo sviluppo di una nuova funzionalità, il ramo principale potrebbe aver ricevuto altre modifiche da diversi sviluppatori o team. Per incorporare queste modifiche nel ramo delle funzionalità, esegui un'`rebase`operazione Git. Questa operazione essenzialmente riproduce i commit del feature branch sopra i commit più recenti del branch principale, assicurando che il feature branch includa tutte le ultime modifiche	Sviluppatore Lex Bot

Operazione	Description	Competenze richieste
Importa e convalida il ticket bot ribasato.	Dopo aver modificato il feature branch, devi importarl o nell'istanza del ticket bot. Questa importazione aggiorna il ticket bot esistente con le ultime modifiche apportate dal ramo rebased. Per importare il ticket bot rebased, attiva la `ImportTic ketBotPipeline` pipeline. La pipeline importa nell'ista nza i file json di definizio ne del ticket bot nel ramo delle funzionalità del sistema di controllo della versione. `TicketBot`	Sviluppatore Lex Bot

Operazione	Description	Competenze richieste
Convalida la definizione del bot ribasata.	Dopo aver importato la definizione del bot ribasato, è fondamentale convalidarne la funzionalità. Vuoi assicurarti che la nuova funzionalità funzioni come previsto e non sia in conflitto con le funzionalità esistenti. Questa convalida prevede in genere il test del bot con vari scenari di input, il controllo delle risposte e la verifica che il bot si comporti come previsto. È possibile eseguire la convalida in uno dei seguenti modi: • Testa il bot manualmente utilizzando la console Amazon Lex. • Utilizza un approccio automatizzato utilizzando framework e strumenti di test in grado di simulare le interazioni degli utenti e affermare le risposte previste.	Sviluppatore Lex Bot

Operazione	Description	Competenze richieste
Unisci il ramo delle funzional ità nel ramo principale.	Dopo aver sviluppato e testato la nuova funzional ità nell'`TicketBot` istanza isolata, effettuate le seguenti operazioni: 1. Applica le modifiche al ramo di funzionalità corrispon dente nel sistema di controllo della versione. 2. Per unire il feature branch al branch principale, create una pull request (PR). Questo PR serve come richiesta per rivedere e incorporare le modifiche nel codebase principale.	Sviluppatore Lex Bot, amministratore del repository

Operazione	Description	Competenze richieste
Elimina il feature branch e il ticket bot.	Dopo che un feature branch è stato unito con successo al branch principale, elimina il feature branch e il ticket bot dal repository del codice sorgente. Per eliminare il feature branch e il ticket bot, attiva la `DeleteTicketBotPipeline` pipeline. La pipeline rimuove le risorse bot temporanee create durante il processo di sviluppo (ad esempio, il ticket bot). Questa azione aiuta a mantenere un repository pulito e a prevenire confusione o conflitti con le future feature branch.	Sviluppatore Lex Bot

Mantieni il bot principale

Operazione	Description	Competenze richieste
Importa l'ultima definizione di bot principale nell'`dev`ambiente.	Per importare nell'`dev`ambiente l'ultima definizione del bot principale nel ramo principale, attiva la `DeployBotDevPipeline` pipeline.	AWS DevOps

Operazione	Description	Competenze richieste
	La pipeline crea anche un tag git al momento dell'approvazione.	
Importa l'ultima definizione di bot principale nell'`prod`ambiente.	Per importare nell'`prod`ambiente l'ultima definizione del bot nel ramo principale, fornite come parametro il riferimento al tag dell'attività precedente e attivate la `DeployBotProdPipeline` pipeline. La pipeline importa l'ultima definizione di bot da un tag specifico nell'`prod`ambiente.	AWS DevOps

risoluzione dei problemi

Problema	Soluzione
Quando distribuisci bot Amazon Lex su diversi account Account AWS, i servizi di tooling devono disporre delle autorizzazioni necessarie per accedere alle risorse di tali account.	Per concedere l'accesso a più account, utilizza i ruoli e le policy IAM. Crea ruoli IAM negli account di destinazione e associa politiche ai ruoli che concedono le autorizzazioni richieste. Quindi, assumi questi ruoli dall'account in cui è distribuito il bot Amazon Lex. Per ulteriori informazioni, consulta le autorizzazioni IAM necessarie per l'importazione e le autorizzazioni IAM necessarie per esportare i bot in Lex V2 nella documentazione di Amazon Lex.

Risorse correlate

- Importazione di bot in Amazon Lex V2
- Avvia una pipeline in CodePipeline
- Utilizzo dei bot Amazon Lex V2

Coordina la dipendenza dalle risorse e l'esecuzione delle attività utilizzando il costrutto AWS Fargate WaitCondition hook

Stan Fan, Amazon Web Services

Riepilogo

Questo modello descrive il pacchetto WaitCondition hook (`waitcondition-hook-for-aws-fargate-task`) npm, una soluzione nativa per il cloud progettata per orchestrare le AWS Fargateattività nei cluster Amazon Elastic Container Service (Amazon ECS) Elastic Container Service (Amazon ECS).

L' WaitCondition hook è un AWS Cloud Development Kit (AWS CDK) costrutto specificamente progettato per l'integrazione con. AWS CloudFormation L' WaitCondition hook offre le seguenti funzionalità chiave:

- Funziona come un meccanismo di attesa, sospendendo l'esecuzione CloudFormation dello stack fino al completamento di un'attività Fargate specificata, il che aiuta con le distribuzioni ordinate e l'approvvigionamento delle risorse.
- Supporti TypeScript e Python, che lo rendono ideale per i AWS CDK progetti.
- Consente a sviluppatori e architetti di orchestrare le implementazioni coordinando il completamento delle attività e la gestione delle risorse per le applicazioni containerizzate su. AWS
- Consente l'esecuzione delle attività di Fargate con uno o più contenitori incorporati in un CloudFormation ciclo di vita e può gestire gli errori delle attività e ripristinare lo stack dopo un errore di attività. CloudFormation
- Fornisce la flessibilità necessaria per aggiungere dipendenze tra le risorse e i risultati dell'esecuzione delle attività di Fargate, abilitando attività personalizzate o richiamando altri endpoint. Ad esempio, è possibile mettere in pausa uno CloudFormation stack e attendere una migrazione del database (eseguita da un'attività di Fargate) e fornire altre risorse che potrebbero dipendere dal successo della migrazione del database.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS

- AWS Cloud Development Kit (AWS CDK) Interfaccia a riga di comando (CLI) installata su una workstation locale. Per ulteriori informazioni, consulta il riferimento alla AWS CDK CLI AWS CDK nella documentazione.
- Node package manager (npm), installato su una workstation locale e configurato per l'ingresso.AWS CDK TypeScript Per ulteriori informazioni, vedere Download e installazione di Node.js e npm nella documentazione di npm.
- Yarn installato su una workstation locale. Per ulteriori informazioni, vedere Installazione nella documentazione di Yarn.

Limitazioni

- Questa soluzione viene distribuita su un unico dispositivo. Account AWS
- Il codice di ritorno previsto `0` per il contenitore indica l'esito positivo. Qualsiasi altro codice restituito indica un errore e lo CloudFormation stack verrà ripristinato.
- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Architecture

Il diagramma seguente mostra l'architettura di costruzione.

Il diagramma mostra il flusso di lavoro di: `waitcondition-hook-for-aws-fargate-task`

1. `WaitCondition`e `WaitConditionHandler` sono predisposti per ascoltare la risposta delle funzioni. AWS Lambda
2. A seconda del risultato dell'attività, il `CallbackFunction` o il `ErrorHandlerFunction` viene attivato dal completamento dell'attività Fargate.
3. La funzione Lambda invia un segnale SUCCEED o FAILURE a. `WaitConditionHandler`
4. `WaitConditionHandler`continua a fornire le risorse se il risultato dell'esecuzione dell'attività Fargate ha esito positivo o ripristina lo stack se l'attività non è riuscita.

Il diagramma seguente mostra un esempio di flusso di lavoro per eseguire una migrazione del database.

Il flusso di lavoro di esempio utilizza il `waitcondition-hook-for-aws-fargate-task` costrutto per eseguire una migrazione del database, come segue:

1. Viene fornita un'istanza Amazon Relational Database Service (Amazon RDS).
2. Il `waitcondition-hook-for-aws-fargate-task` costrutto esegue l'attività di migrazione del database e mette in pausa lo stack come istanza Amazon Elastic Compute Cloud (Amazon EC2).
3. Se l'attività di migrazione viene completata correttamente, invia un segnale Succeed a. CloudFormation Altrimenti, invia un segnale Fail CloudFormation e ripristina lo stack.

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che consente di definire l'infrastruttura cloud in codice e di eseguirne il provisioning. CloudFormation
- CloudFormationti aiuta a configurare AWS le risorse, a fornirle in modo rapido e coerente e a gestirle durante tutto il loro ciclo di vita tra Account AWS e. Regioni AWS
- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue AWS risorse e delle applicazioni su cui esegui AWS in tempo reale.
- Amazon Elastic Container Service (Amazon ECS) è un servizio rapido e scalabile di gestione dei container che ti aiuta a eseguire, arrestare e gestire container in un cluster.
- AWS Fargateti aiuta a eseguire contenitori senza dover gestire server o istanze Amazon EC2. Viene utilizzato in combinazione con Amazon ECS.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS Step Functionsè un servizio di orchestrazione senza server che consente di combinare AWS Lambda funzioni e altro Servizi AWS per creare applicazioni aziendali critiche.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una rete tradizionale che gestiresti nel tuo data center, con i vantaggi dell'utilizzo dell'infrastruttura scalabile di. AWS

Altri strumenti

- npm è un registro software che viene eseguito in un ambiente Node.js e viene utilizzato per condividere o prendere in prestito pacchetti e gestire la distribuzione di pacchetti privati.
- Yarn è un gestore di pacchetti open source che puoi utilizzare per gestire le dipendenze nei progetti. JavaScript Yarn può aiutarti a installare, aggiornare, configurare e rimuovere le dipendenze dei pacchetti.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub waitcondition-hook-for-aws-fargate-task.

Best practice

- Quando crei la tua AWS CDK app, segui le migliori pratiche per lo sviluppo e la distribuzione dell'infrastruttura cloud con la documentazione contenuta nella v2. AWS CDK AWS CDK
- Per AWS Fargate questo compito, segui le best practice per le immagini dei container Amazon ECS nella documentazione di Amazon ECS.

Epiche

Configura il AWS CDK

Operazione	Description	Competenze richieste
Installa il AWS CDK.	Per installarlo AWS CDK sul computer locale o in un altro ambiente, esegui il seguente comando: `npm install -g aws-cdk@latest`	Architetto del cloud, sviluppatore di app
Bootstrap il. AWS CDK	Il bootstrap è il processo di preparazione di un ambiente per la distribuzione. Per	Architetto del cloud

Operazione	Description	Competenze richieste
	avviare il AWS CDK toolkit per il target Account AWS e Regione AWS, esegui il seguente comando: `cdk bootstrap aws://ACCOUNT-NUMBER-1/REGION-1` Questo comando crea uno CloudFormation stack denominato. `CDKToolkit`	

Esegui l' WaitCondition hook for AWS Fargate tasks construct

Operazione	Description	Competenze richieste
Crea il progetto CDK.	Crea un progetto CDK usando il linguaggio che preferisci. Questo modello utilizza TypeScript. Per creare un progetto CDK utilizzando TypeScript, esegui il seguente comando: `cdk init app –language typescript`	Architetto cloud
Installare il pacchetto .	Esegui `npm install` sul percorso principale del tuo progetto CDK. Dopo aver installato la libreria CDK, esegui il seguente comando per installarla: `waitcondi`	Architetto del cloud

Operazione	Description	Competenze richieste

	`tion-hook-for-aws-fargate-task`
	
	`yarn add waitcondition-hook-for-aws-fargate-task`	

Operazione	Description	Competenze richieste
Crea la tua applicazione CDK e i componenti Amazon ECS.	Crea il tuo progetto CDK. È richiesta una risorsa di definizione delle attività di Amazon ECS. Per informazioni sulla creazione di una definizione di attività, consulta le definizioni delle attività di Amazon ECS nella documentazione di Amazon ECS. L'esempio seguente utilizza questo costrutto: `import * as cdk from 'aws-cdk-lib'; import { Vpc } from 'aws-cdk-lib/aws-ec2'; import * as ecr from 'aws-cdk-lib/aws-ecr'; import * as ecs from 'aws-cdk-lib/aws-ecs'; import { Construct } from 'constructs'; import { FargateRunner } from 'waitcondition-hook-for-aws-fargate-task'; import { Queue } from 'aws-cdk-lib/aws-sqs';`  `export class FargateRunnerStack extends cdk.Stack { constructor(scope: Construct, id: string,`	Architetto del cloud

Operazione	Description	Competenze richieste

	props?: cdk.StackProps) {
	super(scope, id, props);
	// Define the VPC
	const vpc = new Vpc(this, 'MyVpc')
	// Define the Fargate Task
	const taskDefinition = new ecs.FargateTaskDefinition(this, 'MyTask', {});
	// Import exiting ecr repo
	const repo = ecr.Repository.fromRepositoryName(this, 'MyRepo', 'RepoName');
	// Add a container to the task
	taskDefinition.addContainer('MyContainer', {
	image: ecs.ContainerImage.fromEcrRepository(repo),
	});
	// Create the Fargate runner
	const myFargateRunner = new FargateRunner(this, 'MyRunner', {
	fargateTaskDef: taskDefinition,
	timeout: `${60 * 5}`,
	vpc: vpc,
	});	

Operazione	Description	Competenze richieste
	`// Create the SQS queue const myQueue = new Queue(this, 'MyQueue', {}); // Add dependency myQueue.node.addDependency(myFargateRunner); } }`	

Sintetizza e avvia l'applicazione CDK.	1. Per generare le risorse e il CloudFormation modello, esegui il seguente comando nel percorso principale del CDK:
	`cdk synth`
	2. Una volta eseguito correttamente il `synth` comando, esegui il comando seguente per distribuire le risorse:
	`cdk deploy`
	
	Il `waitcondition-hook-for-aws-fargate-task` costrutto esegue il task Fargate.
		Architetto del cloud

Eliminazione

Operazione	Description	Competenze richieste
Eliminare le risorse.	Per ripulire le risorse fornite nel passaggio precedente, esegui il comando seguente: `cdk destroy`	Architetto del cloud

Risoluzione dei problemi

Problema	Soluzione
Guasto generale CloudFormation dello stack	Per facilitare la risoluzione dei problemi generali CloudFormation dello stack, aggiungete il `--no-rollback` flag come mostrato nell'esempio seguente: `cdk deploy --no-rollback` Questo comando interromperà il rollback dello CloudFormation stack, il che offre risorse per la risoluzione dei problemi. Per ulteriori informazioni, consulta Scegliere come gestire gli errori durante il provisioning delle risorse nella documentazione. CloudFormation
AWS Step Functions fallimento	Una macchina a AWS Step Functions stati potrebbe non riuscire a funzionare per diversi motivi. Una volta `–disable-rollback` configurato, utilizza i seguenti passaggi per risolvere i problemi:

Problema	Soluzione
	1. Accedi a Console di gestione AWS, inserisci Step Functions nel campo di ricerca, quindi scegli il servizio Step Functions. 2. Nel riquadro di navigazione a sinistra, scegli Macchine a stati, quindi seleziona la macchina a stati fornita dallo CloudForm ation stack. 3. In Esecuzioni, scegli il nome dell'esecuzione che non è riuscita in modo imprevisto. 4. Nella vista Evento, scegliete il passaggio fallito. Per ulteriori informazioni, consulta Risoluzio ne dei problemi in Step Functions e Visualizz azione dei dettagli di esecuzione nella console Step Functions nella AWS Step Functions documentazione.

Problema	Soluzione
AWS Lambda errore della funzione	Questo costrutto fornisce due funzioni `CallbackFunction` Lambda: e. `ErrorhandlerFunction` Possono fallire per vari motivi, ad esempio eccezioni non gestite. Utilizza i seguenti passaggi per risolvere i problemi: 1. Accedi a Console di gestione AWS, inserisci CloudWatchil campo Cerca, quindi scegli il CloudWatch servizio. 2. Nel pannello di navigazione a sinistra, scegli Log groups (Gruppi di log). 3. Nel campo Cerca, inserisci il nome della funzione Lambda. 4. Scegli il nome del gruppo di log associato alla funzione Lambda. 5. Per andare al risultato dell'esecuzione della funzione Lambda, scegli il flusso di log più recente. Per ulteriori informazioni, consulta Risoluzione dei problemi in Lambda nella AWS Lambda documentazione.

Risorse correlate

AWS documentazione

- AWS CDK Costruisci un riferimento all'API
- Iniziare con AWS CDK
- Scopri come creare e utilizzare le risorse Amazon ECS
- Scopri come iniziare a usare Step Functions
- Che cos'è AWS CDK?

Altre risorse

- Waitcondition Hook for AWS Fargate task (npm)
- waitcondition-hook-for-aws-fargate-task 1.0.6 (pypi.org)

Usa repository di sorgenti Git di terze parti in AWS CodePipeline

Kirankumar Chandrashekar, Amazon Web Services

Riepilogo

Questo modello descrive come utilizzare AWS CodePipeline con repository di sorgenti Git di terze parti.

AWS CodePipeline è un servizio di distribuzione continua che automatizza le attività di creazione, test e distribuzione del software. Il servizio attualmente supporta repository Git gestiti da GitHub AWS e CodeCommit Atlassian Bitbucket. Tuttavia, alcune aziende utilizzano repository Git di terze parti integrati con il loro servizio Single Sign-On (SSO) e Microsoft Active Directory per l'autenticazione. Puoi usare questi repository Git di terze parti come fonti CodePipeline per creare azioni e webhook personalizzati.

Un webhook è una notifica HTTP che rileva gli eventi in un altro strumento, ad esempio un GitHub repository, e collega tali eventi esterni a una pipeline. Quando crei un webhook in CodePipeline, il servizio restituisce un URL che puoi usare nel webhook del tuo repository Git. Se invii codice a un ramo specifico del repository Git, il webhook Git avvia il CodePipeline webhook tramite questo URL e imposta la fase di origine della pipeline su In corso. Quando la pipeline è in questo stato, un job worker esegue un sondaggio CodePipeline per il lavoro personalizzato, lo esegue e invia uno stato di successo o di fallimento a. CodePipeline In questo caso, poiché la pipeline si trova nella fase di origine, il job worker ottiene il contenuto del repository Git, lo comprime e lo carica nel bucket Amazon Simple Storage Service (Amazon S3) dove sono archiviati gli elementi per la pipeline, utilizzando la chiave oggetto fornita dal job sottoposto a polling. Puoi anche associare una transizione per l'azione personalizzata a un evento in Amazon CloudWatch e avviare il job worker in base all'evento. Questa configurazione consente di utilizzare repository Git di terze parti che il servizio non supporta nativamente come sorgenti. CodePipeline

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un repository Git che supporta i webhook e può connettersi a un URL di CodePipeline webhook tramite Internet
- AWS Command Line Interface (AWS CLI) installata e configurata per funzionare con l'account AWS

Architecture

Il modello prevede le seguenti fasi:

1. L'utente inserisce il codice in un repository Git.
2. Viene chiamato il webhook Git.
3. Viene chiamato il CodePipeline webhook.
4. La pipeline è impostata su In corso e la fase di origine è impostata sullo stato In corso.
5. L'azione della fase di origine avvia una regola CloudWatch Events, indicando che è stata avviata.
6. L' CloudWatch evento avvia una funzione Lambda.
7. La funzione Lambda ottiene i dettagli del job di azione personalizzato.
8. La funzione Lambda avvia CodeBuild AWS e gli trasmette tutte le informazioni relative al lavoro.
9. CodeBuild ottiene la chiave SSH pubblica o le credenziali utente per l'accesso HTTPS Git da Secrets Manager.
10. CodeBuild clona il repository Git per un ramo specifico.
11. CodeBuild comprime l'archivio e lo carica nel bucket S3 che funge da archivio degli artefatti. CodePipeline

Tools (Strumenti)

- AWS CodePipeline: AWS CodePipeline è un servizio di distribuzione continua completamente gestito che ti aiuta ad automatizzare le pipeline di rilascio per aggiornamenti rapidi e affidabili di applicazioni e infrastrutture. CodePipeline automatizza le fasi di compilazione, test e distribuzione del processo di rilascio per ogni modifica del codice, in base al modello di rilascio definito. Ciò consente di fornire funzionalità e aggiornamenti in modo rapido e affidabile. Puoi integrare AWS CodePipeline con servizi di terze parti come GitHub o con il tuo plug-in personalizzato.
- AWS Lambda: AWS Lambda consente di eseguire codice senza effettuare il provisioning o la gestione di server. Con Lambda, puoi eseguire codice praticamente per qualsiasi tipo di applicazione o servizio di backend senza necessità di amministrazione. Tu carichi il codice e Lambda si occupa di tutto il necessario per eseguire e scalare il codice con un'elevata disponibilità. Puoi configurare il codice in modo che venga avviato automaticamente da altri servizi AWS o chiamarlo direttamente da qualsiasi app Web o mobile.

- AWS CodeBuild: AWS CodeBuild è un servizio di integrazione continua completamente gestito che compila codice sorgente, esegue test e produce pacchetti software pronti per la distribuzione. Con CodeBuild, non è necessario fornire, gestire e scalare i propri server di build. CodeBuild esegue la scalabilità continua ed elabora più build contemporaneamente, in modo che le build non restino in attesa in coda. Puoi iniziare a utilizzare CodeBuild velocemente con ambienti di compilazione predefiniti oppure puoi creare ambienti di compilazione personalizzati che utilizzano strumenti di compilazione specifici.
- AWS Secrets Manager — AWS Secrets Manager ti aiuta a proteggere i segreti necessari per accedere alle tue applicazioni, servizi e risorse IT. Il servizio consente di ruotare, gestire e recuperare le credenziali del database, le chiavi API e altri segreti durante tutto il loro ciclo di vita. Gli utenti e le applicazioni recuperano i segreti chiamando Secrets Manager APIs, senza dover codificare le informazioni sensibili in testo normale. Secrets Manager offre una rotazione segreta con integrazione integrata per Amazon Relational Database Service (Amazon RDS), Amazon Redshift e Amazon DocumentDB. Il servizio può essere esteso per supportare altri tipi di segreti, tra cui chiavi e token API. OAuth Inoltre, Secrets Manager consente di controllare l'accesso ai segreti utilizzando autorizzazioni granulari e di controllare centralmente la rotazione segreta per le risorse nel cloud AWS, nei servizi di terze parti e negli ambienti locali.
- Amazon CloudWatch: Amazon CloudWatch è un servizio di monitoraggio e osservazione creato per DevOps ingegneri, sviluppatori, ingegneri dell'affidabilità del sito (SREs) e responsabili IT. CloudWatch ti fornisce dati e approfondimenti utilizzabili per monitorare le tue applicazioni, rispondere ai cambiamenti delle prestazioni a livello di sistema, ottimizzare l'utilizzo delle risorse e ottenere una visione unificata dello stato operativo. CloudWatch raccoglie dati operativi e di monitoraggio sotto forma di log, metriche ed eventi, fornendo una visione unificata delle risorse, delle applicazioni e dei servizi AWS eseguiti su server AWS e locali. Puoi utilizzarli CloudWatch per rilevare comportamenti anomali nei tuoi ambienti, impostare allarmi, visualizzare log e metriche fianco a fianco, intraprendere azioni automatizzate, risolvere problemi e scoprire approfondimenti per far funzionare le tue applicazioni senza intoppi.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti che consente di archiviare e proteggere qualsiasi quantità di dati per una vasta gamma di casi d'uso, come siti Web, applicazioni mobili, backup e ripristino, archiviazione, applicazioni aziendali, dispositivi IoT e analisi di big data. Amazon S3 offre funzionalità di easy-to-use gestione per aiutarti a organizzare i dati e configurare controlli di accesso ottimizzati per soddisfare requisiti aziendali, organizzativi e di conformità specifici.

Epiche

Crea un'azione personalizzata in CodePipeline

Operazione	Description	Competenze richieste
Crea un'azione personalizzata utilizzando AWS CLI o AWS. CloudFormation	Questo passaggio prevede la creazione di un'azione sorgente personalizzata che può essere utilizzata nella fase di origine di una pipeline nel tuo account AWS in una particolare regione. È necessario utilizzare AWS CLI o AWS CloudFormation (non la console) per creare l'azione di origine personali zzata. Per ulteriori informazi oni sui comandi e i passaggi descritti in questa e in altre epopee, consulta la sezione «Risorse correlate» alla fine di questo schema. Nella CLI di AWS, usa il create-custom-action-type comando. Usa --configuration-properties per fornire tutti i parametri richiesti al job worker da elaborare quando effettua un sondaggio per un lavoro. CodePipeline Assicurati di annotare i valori forniti alle opzioni --provide r e --action-version, in modo da poter utilizzare gli stessi valori durante la creazione della pipeline con questa fase di origine personalizzata.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	Puoi anche creare l'azione sorgente personalizzata in AWS CloudFormation utilizzando il tipo di risorsa AWS::CodePipeline::CustomActionType.	

Configurare l’autenticazione

Operazione	Description	Competenze richieste
Creare una coppia di chiavi SSH.	Crea una coppia di key pair Secure Shell (SSH). Per istruzioni, consultate la GitHub documentazione.	Sistemi/ ingegnere DevOps
Crea un segreto in AWS Secrets Manager.	Copia il contenuto della chiave privata dalla coppia di chiavi SSH e crea un segreto in AWS Secrets Manager. Questo segreto viene utilizzato per l'autenticazione quando si accede al repository Git.	Informazioni generali su AWS
Aggiungi la chiave pubblica al repository Git.	Aggiungi la chiave pubblica dalla coppia di chiavi SSH alle impostazioni dell'account del repository Git, per l'autenticazione con la chiave privata.	Sistemi/ ingegnere DevOps

Crea una pipeline e un webhook

Operazione	Description	Competenze richieste
Crea una pipeline che includa l'azione di origine personalizzata.	Crea una pipeline in. CodePipeline Quando configuri la fase di origine, scegli l'azione di origine personalizzata che hai creato in precedenza. Puoi farlo nella CodePipeline console AWS o nell'interfaccia a riga di comando di AWS. CodePipeline richiede le proprietà di configurazione impostate nell'azione personalizzata. Queste informazioni sono necessarie affinché il job worker elabori il lavoro per l'azione personalizzata. Segui la procedura guidata e crea la fase successiva per la pipeline.	Informazioni generali su AWS
Crea un CodePipeline webhook.	Crea un webhook per la pipeline che hai creato con l'azione source personalizzata. È necessario utilizzare AWS CLI o AWS CloudFormation (non la console) per creare il webhook. Nella CLI di AWS, esegui il comando put-webhook e fornisci i valori appropriati per le opzioni webhook. Prendi nota dell'URL del webhook restituito dal comando. Se utilizzi AWS CloudFormation	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	per creare il webhook, usa il tipo di AWS::CodePipeline: :Webhook risorsa. Assicurati di generare l'URL del webhook dalla risorsa creata e prendine nota.	

Operazione	Description	Competenze richieste
Crea una funzione e CodeBuild un progetto Lambda.	In questo passaggio, si utilizza Lambda CodeBuild per creare un job worker che analizzerà le richieste di lavoro CodePipeline per l'azione personalizzata, eseguirà il lavoro e restituirà il risultato dello stato a. CodePipeline Crea una funzione Lambda che viene avviata da una regola di Amazon CloudWatch Events quando la fase di azione sorgente personalizzata della pipeline passa a «In corso». Quando viene avviata, la funzione Lambda dovrebbe ottenere i dettagli dell'azione personalizzata eseguendo il polling dei lavori. Puoi utilizzare l' PollForJobs API per restituire queste informazioni. Dopo aver ottenuto le informazioni sul lavoro oggetto del sondaggio, la funzione Lambda dovrebbe restituire un riconoscimento e quindi elaborare le informazioni con i dati ottenuti dalle proprietà di configurazione per l'azione personalizzata. Quando l'operatore è pronto a parlare con il repository Git, potresti avviare un CodeBuild progetto, perché è comodo	General AWS, sviluppatore di codice

Operazione	Description	Competenze richieste
	gestire le attività Git utilizzan do il client SSH.	

Crea un evento in CloudWatch

Operazione	Description	Competenze richieste
Crea una regola per CloudWatch gli eventi.	Crea una regola CloudWatch Events che avvia la funzione Lambda come destinazione ogni volta che la fase di azione personalizzata della pipeline passa a «In corso».	Informazioni generali su AWS

Risorse correlate

Creazione di un'azione personalizzata in CodePipeline

- Creare e aggiungere un'azione personalizzata in CodePipeline
- AWS::CodePipeline::CustomActionType risorsa

Configurazione dell'autenticazione

- Creazione e gestione di segreti con AWS Secrets Manager

Creazione di una pipeline e di un webhook

- Crea una pipeline in CodePipeline
- riferimento al comando put-webhook
- AWS::CodePipeline::Webhook risorsa
- PollForJobs Documentazione di riferimento dell'API
- Crea e aggiungi un'azione personalizzata in CodePipeline
- Crea un progetto di compilazione in AWS CodeBuild

Creare un evento

- Rileva e reagisci ai cambiamenti nello stato della pipeline con Amazon Events CloudWatch

Riferimenti aggiuntivi

- Lavorare con le condutture in CodePipeline
- Guida per sviluppatori AWS Lambda

Crea una CI/CD pipeline per convalidare le configurazioni Terraform utilizzando AWS CodePipeline

Aromal Raj Jayarajan e Vijesh Vijayakumaran Nair, Amazon Web Services

Riepilogo

Questo modello mostra come testare le configurazioni HashiCorp Terraform utilizzando una pipeline di integrazione e distribuzione continua (CI/CD) distribuita da AWS. CodePipeline

Terraform è un'applicazione di interfaccia a riga di comando che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud. La soluzione fornita in questo modello crea una CI/CD pipeline che ti aiuta a convalidare l'integrità delle tue configurazioni Terraform eseguendo cinque fasi: CodePipeline

1. `"checkout"`estrae la configurazione Terraform che stai testando da un repository CodeCommit AWS.
2. `"validate"`esegue strumenti di convalida dell'infrastruttura come codice (IaC), inclusi tfsec e checkov. TFLint Lo stage esegue anche i seguenti comandi di convalida Terraform IAc: e. `terraform validate` `terraform fmt`
3. `"plan"`mostra quali modifiche verranno applicate all'infrastruttura se viene applicata la configurazione Terraform.
4. `"apply"`utilizza il piano generato per fornire l'infrastruttura richiesta in un ambiente di test.
5. `"destroy"`rimuove l'infrastruttura di test creata durante la `"apply"` fase.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- AWS Command Line Interface (AWS CLI), installata e configurata
- Git, installato e configurato sul tuo computer locale
- Terraform, installato e configurato sul computer locale

Limitazioni

- L'approccio di questo modello implementa AWS CodePipeline in un solo account AWS e in una sola regione AWS. Le modifiche alla configurazione sono necessarie per le distribuzioni con più account e più regioni.
- Il ruolo AWS Identity and Access Management (IAM) fornito da questo modello (codepipeline_iam_role) segue il principio del privilegio minimo. Le autorizzazioni di questo ruolo IAM devono essere aggiornate in base alle risorse specifiche che la pipeline deve creare.

Versioni del prodotto

- AWS CLI versione 2.9.15 o successiva
- Terraform versione 1.3.7 o successiva

Architecture

Stack tecnologico Target

- AWS CodePipeline
- AWS CodeBuild
- AWS CodeCommit
- AWS IAM
- Amazon Simple Storage Service (Amazon S3)
- AWS Key Management Service (AWS KMS)
- Terraform

Architettura Target

Il diagramma seguente mostra un esempio di flusso di lavoro della CI/CD pipeline per testare le configurazioni Terraform. CodePipeline

Il diagramma mostra il flusso di lavoro seguente:

1. Nel CodePipeline, un utente AWS avvia le azioni proposte in un piano Terraform eseguendo il `terraform apply` comando nella CLI AWS.
2. AWS CodePipeline assume un ruolo di servizio IAM che include le policy necessarie per l'accesso CodeCommit CodeBuild, AWS KMS e Amazon S3.

3. CodePipeline esegue la fase della `"checkout"` pipeline per estrarre la configurazione Terraform da un CodeCommit repository AWS per i test.
4. CodePipeline esegue la `"validate"` fase per testare la configurazione Terraform eseguendo strumenti di convalida IaC ed eseguendo comandi di convalida Terraform IAc in un progetto. CodeBuild
5. CodePipeline esegue la `"plan"` fase per creare un piano nel CodeBuild progetto basato sulla configurazione Terraform. L'utente AWS può rivedere questo piano prima che le modifiche vengano applicate all'ambiente di test.
6. Code Pipeline esegue la `"apply"` fase di implementazione del piano utilizzando il CodeBuild progetto per fornire l'infrastruttura richiesta nell'ambiente di test.
7. CodePipeline esegue lo `"destroy"` stage, che utilizza CodeBuild per rimuovere l'infrastruttura di test creata durante la `"apply"` fase.
8. Un bucket Amazon S3 archivia gli elementi della pipeline, che vengono crittografati e decrittografati utilizzando una chiave gestita dal cliente AWS KMS.

Tools (Strumenti)

Strumenti

Servizi AWS

- AWS ti CodePipeline aiuta a modellare e configurare rapidamente le diverse fasi di un rilascio di software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- AWS CodeBuild è un servizio di build completamente gestito che ti aiuta a compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS CodeCommit è un servizio di controllo delle versioni che consente di archiviare e gestire archivi Git in modo privato, senza dover gestire il proprio sistema di controllo del codice sorgente.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri servizi

- HashiCorp Terraform è un'applicazione di interfaccia a riga di comando che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud.

Codice

Il codice per questo pattern è disponibile nel repository. GitHub aws-codepipeline-terraform-cicdsamples Il repository contiene le configurazioni Terraform necessarie per creare l'architettura di destinazione delineata in questo modello.

Epiche

Fornisci i componenti della soluzione

Operazione	Description	Competenze richieste
Clona il GitHub repository.	Clona il GitHub aws-codepipeline-terraform-cicdsamplesrepository eseguendo il seguente comando in una finestra di terminale: `git clone https://github.com/aws-samples/aws-codepipeline-terraform-cicd-samples.git` Per ulteriori informazioni, consulta Clonazione di un repository nella documentazione. GitHub	DevOps ingegnere
Crea un file di definizioni variabili Terraform.	Crea un `terraform.tfvars` file in base ai requisiti del tuo caso d'uso. Puoi aggiornare le variabili nel `examples/terraform`	DevOps ingegnere

Operazione	Description	Competenze richieste
	`.tfvars` file che si trova nel repository clonato. Per ulteriori informazioni, consulta Assegnazione di valori alle variabili del modulo root nella documentazione di Terraform. **Note** Il `Readme.md` file del repository include ulteriori informazioni sulle variabili richieste.	
Configura AWS come provider Terraform.	1. In un editor di codice, apri il file del repository clonato. `main.tf` 2. Aggiungi le configurazioni necessarie per stabilire la connettività all'account AWS di destinazione. Per ulteriori informazioni, consulta il provider AWS nella documentazione di Terraform.	DevOps ingegnere

Operazione	Description	Competenze richieste
Aggiorna la configurazione del provider Terraform per creare il bucket di replica Amazon S3.	1. Apri la `S3` directory del repository eseguendo il seguente comando: `cd ./modules/s3` 2. Aggiorna la configurazione del provider Terraform per creare il bucket di replica Amazon S3 aggiornando `region` il valore nel file. `tf` Assicurati di inserire la regione in cui desideri che Amazon S3 replichi gli oggetti. 3. (Facoltativo) Per impostazi one predefinita, Terraform utilizza file di stato locali per la gestione dello stato. Se desideri aggiungere Amazon S3 come backend remoto, devi aggiornare la configurazione di Terraform . Per ulteriori informazioni, consulta Configurazione del backend nella documenta zione di Terraform. **Note** La replica attiva la copia automatica e asincrona degli oggetti	DevOps ingegnere

Operazione	Description	Competenze richieste
	tra i bucket Amazon S3.	
Inizializza la configurazione Terraform.	Per inizializzare la directory di lavoro che contiene i file di configurazione Terraform, esegui il seguente comando nella cartella principale del repository clonato: `terraform init`	DevOps ingegnere
Crea il piano Terraform.	Per creare un piano Terraform , esegui il seguente comando nella cartella principale del repository clonato: `terraform plan --var-file=terraform.tfvars -out=tfplan` **Note** Terraform valuta i file di configurazione per determinare lo stato di destinazione per le risorse dichiarate. Quindi confronta lo stato di destinazione con lo stato attuale e crea un piano.	DevOps ingegnere

Operazione	Description	Competenze richieste
Verifica il piano Terraform.	Rivedi il piano Terraform e conferma che configuri l'architettura richiesta nel tuo account AWS di destinazione.	DevOps ingegnere
Distribuire la soluzione.	1. Per applicare il piano Terraform, esegui il seguente comando nella cartella principale del repository clonato: `terraform apply "tfplan"` 2. Inserisci yes per confermare che desideri distribuire le risorse. **Note** Terraform crea, aggiorna o distrugge l'infrastruttura per raggiungere lo stato di destinazione dichiarato nei file di configurazione.	DevOps ingegnere

Convalida le configurazioni Terraform eseguendo la pipeline

Operazione	Description	Competenze richieste
Configura il repository del codice sorgente.	1. Dall'output di Terraform, ottieni i dettagli del repositor	DevOps ingegnere

Operazione	Description	Competenze richieste
	y di origine per il repositor y che contiene le configura zioni Terraform che desideri convalidare. 2. Accedi alla Console di gestione AWS. Quindi, apri la console. CodeCommit 3. Crea un nuovo ramo nel repository di origine denominato`main`. Per istruzioni, consulta Creare un ramo CodeCommit in AWS nella CodeCommit documentazione. 4. Clona il `main` ramo del repository di origine sulla tua workstation locale. Per istruzioni, consulta i passaggi di configura zione per le connessioni HTTPS ai CodeCommi t repository AWS su Windows con l'aiuto per le credenziali dell'interfaccia a riga di comando AWS nella documentazione. CodeCommit 5. Copia la **`templates`** cartella dal GitHub aws-codepipeline-terraform-cicdsamplesrepository eseguendo il seguente comando:	

Operazione	Description	Competenze richieste
	`cp -r templates $YOUR_CODECOMMIT_REPO_ROOT` **Note** La `templates` cartella contiene i file delle specifiche di build e lo script di convalida per la directory principale del repository di origine. 6. Aggiungi le configurazioni Terraform IAc richieste alla cartella principale del repository di origine. 7. Aggiungi i dettagli per il backend remoto nella configurazione Terraform del tuo progetto. Per ulteriori informazioni, consulta S3 nella documentazione di Terraform. 8. (Facoltativo) Aggiorna le variabili nella `templates` cartella per attivare o disattivare le scansioni preconfigurate, le versioni di modifica degli strumenti e per specificare la directory	

Operazione	Description	Competenze richieste
	nei file di script personali zzati. Per ulteriori informazi oni, consultate la sezione Informazioni aggiuntive di questo modello. 9. Invia le modifiche al `main` ramo del repository di origine.	

Operazione	Description	Competenze richieste
Convalida le fasi della pipeline.	1. Accedi alla console di gestione AWS e apri la console CodePipeline . 2. Nell'output generato dal `terraform apply "tfplan"` comando nella sezione precedente di Epic, trova il nome del comando generato. CodePipeline 3. Apri la pipeline nella CodePipeline console e scegli Release change. 4. Esamina ogni fase della pipeline e conferma che funzioni come previsto. Per ulteriori informazioni, consulta Visualizza i dettagli e la cronologia della pipeline (console) nella AWS CodePipeline User Guide. **Important** Quando viene apportata una modifica al ramo principale del repository di origine, la pipeline di test viene attivata automaticamente.	DevOps ingegnere

Operazione	Description	Competenze richieste
Verifica l'output del rapporto.	1. Sulla CodePipeline console, nel riquadro di navigazione a sinistra, scegli Crea. Quindi, scegli Segnala cronologia. 2. Esamina i report di scansione tfsec e checkov generati dalla pipeline. Questi report possono aiutarti a identificare i problemi tramite visualizzazioni e rappresentazioni grafiche. **Note** Il `<project_name>-validate` CodeBuild progetto genera report di vulnerabilità per il codice durante la fase. `"validate"`	DevOps ingegnere

Pulizia delle risorse

Operazione	Description	Competenze richieste
Pulisci la pipeline e le risorse associate.	Per eliminare le risorse di test dal tuo account AWS, esegui il seguente comando nella cartella principale del repository clonato:	DevOps ingegnere

Operazione	Description	Competenze richieste
	`terraform destroy --var-file=terraform.tfvars`	

risoluzione dei problemi

Problema	Soluzione
Riceverai un AccessDenied errore durante lo `"apply"` stage.	1. Esamina i log di esecuzione del CodeBuild progetto associato alla `"apply"` fase per identificare eventuali autorizzazioni IAM mancanti. Per ulteriori informazioni, consulta Visualizza i dettagli della build in AWS CodeBuild nella AWS CodeBuild User Guide. 2. In un editor di codice, apri la cartella del repository clonato. `modules` Quindi, vai alla `iam-role` cartella e apri il `main.tf` file che si trova in quella cartella. 3. Nella `codepipeline_policy` dichiarazione, aggiungi le policy IAM necessarie per il provisioning delle risorse nel tuo account AWS.

Risorse correlate

- Blocchi di moduli (documentazione Terraform)
- Come utilizzare CI/CD per distribuire e configurare i servizi di sicurezza AWS con Terraform (post sul blog AWS)
- Utilizzo di ruoli collegati ai servizi (documentazione IAM)
- create-pipeline (documentazione AWS CLI)
- Configurare la crittografia lato server per gli artefatti archiviati in Amazon S3 per (documentazione AWS) CodePipeline CodePipeline

- Quote per AWS CodeBuild (CodeBuild documentazione AWS)
- Protezione dei dati in AWS CodePipeline (CodePipeline documentazione AWS)

Informazioni aggiuntive

Moduli Terraform personalizzati

Di seguito è riportato un elenco di moduli Terraform personalizzati utilizzati in questo modello:

- `codebuild_terraform`crea i CodeBuild progetti che formano ogni fase della pipeline.
- `codecommit_infrastructure_source_repo`acquisisce e crea il repository di origine CodeCommit .
- `codepipeline_iam_role`crea i ruoli IAM richiesti per la pipeline.
- `codepipeline_kms`crea la chiave AWS KMS richiesta per la crittografia e la decrittografia degli oggetti Amazon S3.
- `codepipeline_terraform`crea la pipeline di test per il repository di origine. CodeCommit
- `s3_artifacts_bucket`crea un bucket Amazon S3 per gestire gli artefatti della pipeline.

Crea file di specifiche

Di seguito è riportato un elenco di file di specifiche di build (buildspec) utilizzati da questo pattern per eseguire ogni fase della pipeline:

- `buildspec_validate.yml`gestisce il palco. `"validate"`
- `buildspec_plan.yml`dirige il `"plan"` palco.
- `buildspec_apply.yml`dirige il `"apply"` palco.
- `buildspec_destroy.yml`dirige il `"destroy"` palco.

Crea variabili del file di specificazione

Ogni file buildspec utilizza le seguenti variabili per attivare diverse impostazioni specifiche della build:

Variabile	Valore predefinito	Description
`CODE_SRC_DIR`	`"."`	CodeCommit Definisce la directory di origine

TF_VERSION	«1.3.7"	Definisce la versione Terraform per l'ambiente di compilazione

Il `buildspec_validate.yml` file supporta anche le seguenti variabili per attivare diverse impostazioni specifiche della build:

Variabile	Valore predefinito	Description
SCRIPT_DIR	». /modelli/script»	Definisce la directory degli script
ENVIRONMENT	«dev»	Definisce il nome dell'ambi ente
SKIPVALIDATIONFAIL URE	«Y»	Salta la convalida in caso di errori
ENABLE_TFVALIDATE	«Y»	Attiva Terraform validate
ENABLE_TFFORMAT	«Y»	Attiva il formato Terraform
ENABLE_TFCHECKOV	«Y»	Attiva la scansione Check-Ov
ENABLE_TFSEC	«Y»	Attiva la scansione tfsec
TFSEC_VERSION	«v1.28.1"	Definisce la versione di tfsec

Altri modelli

- Accedi alle applicazioni container in modo privato su Amazon EKS utilizzando AWS PrivateLink e un Network Load Balancer
- Associa un AWS CodeCommit repository in uno Account AWS ad Amazon SageMaker AI Studio Classic in un altro account
- Automatizza la creazione di account utilizzando Landing Zone Accelerator su AWS
- Automatizza l'aggiunta o l'aggiornamento delle voci di registro di Windows utilizzando AWS Systems Manager
- Automatizza i backup per le istanze DB di Amazon RDS for PostgreSQL utilizzando AWS Batch
- Automatizza la distribuzione di applicazioni annidate utilizzando AWS SAM
- Automatizza la distribuzione di Node Termination Handler in Amazon EKS utilizzando una pipeline CI/CD
- Automatizza la configurazione di RabbitMQ in Amazon MQ
- Automatizza la replica delle istanze Amazon RDS su Account AWS
- Crea e distribuisci automaticamente un'applicazione Java su Amazon EKS utilizzando una pipeline CI/CD
- Genera automaticamente un modello PynamoDB e funzioni CRUD per Amazon DynamoDB utilizzando un'applicazione Python
- Convalida e distribuisci automaticamente le policy e i ruoli IAM utilizzando CodePipeline IAM Access Analyzer e le macro AWS CloudFormation
- Eseguire il backup dei server Sun SPARC nell'emulatore Stromasys Charon-SSP sul Cloud AWS
- Crea una pipeline di dati per importare, trasformare e analizzare i dati di Google Analytics utilizzando il AWS DataOps Development Kit
- Create un Micro Focus Enterprise Server PAC con Amazon EC2 Auto Scaling e Systems Manager
- Crea una pipeline per immagini di container rinforzate utilizzando Image EC2 Builder e Terraform
- Crea un MLOps flusso di lavoro usando Amazon SageMaker AI e Azure DevOps
- Centralizza la risoluzione DNS utilizzando AWS Managed Microsoft AD e localmente Microsoft Active Directory
- Pulisci le risorse di AWS Account Factory for Terraform (AFT) in modo sicuro dopo la perdita dei file di stato
- Configura la registrazione per le applicazioni.NET in Amazon CloudWatch Logs utilizzando NLog

- Copia le immagini dei container Amazon ECR su e Account AWS Regioni AWS
- Crea un'immagine di contenitore Docker personalizzata SageMaker e usala per l'addestramento dei modelli in AWS Step Functions
- Crea una pipeline nelle regioni AWS che non supportano AWS CodePipeline
- Crea allarmi per metriche personalizzate utilizzando il rilevamento delle anomalie di Amazon CloudWatch
- Personalizza i nomi dei ruoli predefiniti utilizzando AWS CDK gli aspetti e gli escape hatch
- Implementa una pipeline che rilevi simultaneamente i problemi di sicurezza in più risultati di codice
- Implementa e gestisci un data lake serverless sul cloud AWS utilizzando l'infrastruttura come codice
- Distribuisci applicazioni containerizzate in AWS IoT Greengrass V2 esecuzione come contenitore Docker
- Distribuisci risorse e pacchetti Kubernetes utilizzando Amazon EKS e un repository di grafici Helm in Amazon S3
- Distribuisci applicazioni multi-stack utilizzando AWS CDK con TypeScript
- Implementa la convalida della sicurezza della codifica in tempo reale utilizzando un server MCP con Kiro e altri assistenti di codifica
- Distribuisci istanze di cluster di failover di SQL Server su Amazon e EC2 Amazon FSx utilizzando Terraform
- Implementa le automazioni di sicurezza per la AWS WAF soluzione utilizzando Terraform
- Sviluppa assistenti avanzati basati sull'intelligenza artificiale generativa utilizzando RAG e suggerimenti ReAct
- Abilita Amazon in GuardDuty modo condizionale utilizzando modelli AWS CloudFormation
- Configura la scalabilità automatica basata sugli eventi in Amazon EKS utilizzando Amazon EKS Pod Identity e KEDA
- Genera consigli personalizzati e riclassificati con Amazon Personalize
- Ricevi notifiche Amazon SNS quando lo stato chiave di una chiave AWS KMS cambia
- Gestisci i set di autorizzazioni per più account utilizzando Account Factory for Terraform
- Identifica automaticamente le immagini duplicate dei container durante la migrazione a un repository Amazon ECR
- Implementa il controllo delle versioni delle API basato sul percorso utilizzando domini personalizzati in Amazon API Gateway

- Migliora le prestazioni operative abilitando Amazon DevOps Guru su più regioni AWS, account e OUs con AWS CDK
- Installa SSM Agent sui nodi di lavoro Amazon EKS utilizzando Kubernetes DaemonSet
- Integra il controller universale Stonebranch con la modernizzazione del mainframe AWS
- Modernizzazione del mainframe: attiva DevOps AWS con Rocket Software Enterprise Suite
- Gestisci i set di AWS IAM Identity Center autorizzazioni come codice utilizzando AWS CodePipeline
- Gestisci i set di AWS autorizzazioni in modo dinamico utilizzando Terraform
- Gestisci le applicazioni container locali configurando Amazon ECS Anywhere con AWS CDK
- Gestisci AWS Organizations le policy come codice utilizzando AWS CodePipeline Amazon Bedrock
- Esegui la migrazione di record DNS in blocco verso una zona ospitata privata di Amazon Route 53
- Esegui la migrazione di applicazioni ospitate da IIS su Amazon utilizzando appcmd.exe EC2
- Monitora l'uso di un'Amazon Machine Image condivisa su più Account AWS
- Orchestra una pipeline ETL con convalida, trasformazione e partizionamento utilizzando AWS Step Functions
- Automatizza le blue/green implementazioni dei database globali di Amazon Aurora utilizzando i principi IaC
- Conserva lo spazio IP instradabile nei progetti VPC multi-account per sottoreti non destinate ai carichi di lavoro
- Fornisci un prodotto Terraform AWS Service Catalog utilizzando un repository di codice
- Esegui le attività di AWS Systems Manager Automation in modo sincrono da AWS Step Functions
- Configura una CI/CD pipeline per carichi di lavoro ibridi su Amazon ECS Anywhere utilizzando AWS CDK e GitLab
- Configura una CI/CD pipeline per la migrazione del database utilizzando Terraform
- Configura un'infrastruttura Multi-AZ per SQL Server Always On FCI utilizzando Amazon FSx
- Configura automaticamente i bot UiPath RPA su Amazon EC2 utilizzando AWS CloudFormation
- Semplifica l'autenticazione delle applicazioni con TLS reciproco in Amazon ECS utilizzando Application Load Balancer
- Onboarding dei tenant nell'architettura SaaS per il modello a silo utilizzando C# e AWS CDK
- Usa Terraform per abilitare automaticamente Amazon GuardDuty per un'organizzazione
- Usa gli agenti Amazon Bedrock per automatizzare la creazione di controlli di accesso in Amazon EKS tramite istruzioni basate su testo

- Convalida il codice Account Factory for Terraform (AFT) localmente
- Visualizza i risultati AI/ML del modello utilizzando Flask e AWS Elastic Beanstalk

Infrastruttura

Argomenti

- Accedi a un host bastion utilizzando Session Manager e Amazon EC2 Instance Connect
- Centralizza la risoluzione DNS utilizzando AWS Managed Microsoft AD e localmente Microsoft Active Directory
- Centralizza il monitoraggio utilizzando Amazon CloudWatch Observability Access Manager
- Verifica la presenza di tag obbligatori EC2 nelle istanze al momento del lancio
- Pulisci le risorse di AWS Account Factory for Terraform (AFT) in modo sicuro dopo la perdita dei file di stato
- Crea una pipeline nelle regioni AWS che non supportano AWS CodePipeline
- Personalizza i nomi dei ruoli predefiniti utilizzando AWS CDK gli aspetti e gli escape hatch
- Implementa un cluster Cassandra su Amazon EC2 con dati statici privati IPs per evitare il ribilanciamento
- Estendi VRFs ad AWS utilizzando AWS Transit Gateway Connect
- Ricevi notifiche Amazon SNS quando lo stato chiave di una chiave AWS KMS cambia
- Conserva lo spazio IP instradabile nei progetti VPC multi-account per sottoreti non destinate ai carichi di lavoro
- Fornisci un prodotto Terraform AWS Service Catalog utilizzando un repository di codice
- Registrazione di più indirizzi e-mail Account AWS con un unico indirizzo e-mail utilizzando Amazon SES
- Configura la risoluzione DNS per reti ibride in un ambiente AWS con account singolo
- Configura automaticamente i bot UiPath RPA su Amazon EC2 utilizzando AWS CloudFormation
- Configura un' PeopleSoft architettura ad alta disponibilità su AWS
- Configura il disaster recovery per Oracle JD Edwards con EnterpriseOne AWS Elastic Disaster Recovery
- Imposta il rilevamento della CloudFormation deriva in un'organizzazione multiregionale e con più account
- Importa con successo un bucket S3 come stack AWS CloudFormation
- Sincronizza i dati tra i file system Amazon EFS in diverse regioni AWS utilizzando AWS DataSync
- AWS Infrastruttura di test utilizzando LocalStack e Terraform Tests
- Aggiornamento dei cluster SAP Pacemaker da a ENSA1 ENSA2

- Usa zone di disponibilità coerenti in VPCs diversi account AWS
- Usa l'utente IDs nelle policy IAM per il controllo e l'automazione degli accessi
- Convalida il codice Account Factory for Terraform (AFT) localmente
- Altri modelli

Accedi a un host bastion utilizzando Session Manager e Amazon EC2 Instance Connect

Piotr Chotkowski e Witold Kowalik, Amazon Web Services

Riepilogo

Un bastion host, a volte chiamato jump box, è un server che fornisce un unico punto di accesso da una rete esterna alle risorse situate in una rete privata. Un server esposto a una rete pubblica esterna, come Internet, rappresenta un potenziale rischio per la sicurezza in caso di accesso non autorizzato. È importante proteggere e controllare l'accesso a questi server.

Questo modello descrive come utilizzare Session Manager e Amazon EC2 Instance Connect per connetterti in modo sicuro a un host bastion di Amazon Elastic Compute Cloud EC2 (Amazon) distribuito nel tuo. Account AWS Session Manager è una funzionalità di. AWS Systems Manager I vantaggi di questo modello includono:

- L'host bastion distribuito non dispone di porte aperte in ingresso esposte alla rete Internet pubblica. Ciò riduce la potenziale superficie di attacco.
- Non è necessario archiviare e mantenere chiavi Secure Shell (SSH) a lungo termine nel tuo Account AWS. Invece, ogni utente genera una nuova coppia di chiavi SSH ogni volta che si connette all'host bastion. AWS Identity and Access Management Le policy (IAM) allegate alle AWS credenziali dell'utente controllano l'accesso all'host bastion.

Destinatari

Questo modello è destinato ai lettori che hanno esperienza con una conoscenza di base di Amazon EC2, Amazon Virtual Private Cloud (Amazon VPC) e Hashicorp Terraform.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- AWS Command Line Interface (AWS CLI) versione 2, installata e configurata
- Plugin Session Manager per AWS CLI, installato
- CLI Terraform, installata

- Storage per lo stato Terraform, ad esempio un bucket Amazon Simple Storage Service (Amazon S3) e una tabella Amazon DynamoDB che funge da backend remoto per archiviare lo stato Terraform. Per ulteriori informazioni sull'utilizzo dei backend remoti per lo stato Terraform, consulta Amazon S3 Backends (documentazione Terraform). Per un esempio di codice che configura la gestione remota dello stato con un backend Amazon S3, consulta remote-state-s3-backend (Terraform Registry). Si notino i requisiti seguenti:
  - Il bucket Amazon S3 e la tabella DynamoDB devono trovarsi nello stesso spazio. Regione AWS
  - Quando si crea la tabella DynamoDB, la chiave di partizione deve `LockID` essere (distinzione tra maiuscole e minuscole) e il tipo di chiave di partizione deve essere. `String` Tutte le altre impostazioni della tabella devono avere i valori predefiniti. Per ulteriori informazioni, consulta Informazioni sulle chiavi primarie e Creazione di una tabella nella documentazione di DynamoDB.
- Un client SSH, installato

Limitazioni

- Questo modello è inteso come proof of concept (PoC) o come base per ulteriori sviluppi. Non deve essere utilizzato nella sua forma attuale in ambienti di produzione. Prima della distribuzione, modifica il codice di esempio nel repository in base ai requisiti e al caso d'uso.
- Questo modello presuppone che l'host bastion di destinazione utilizzi Amazon Linux 2 come sistema operativo. Sebbene sia possibile utilizzare altre Amazon Machine Images (AMIs), altri sistemi operativi non rientrano nell'ambito di questo schema.

> **Note**
>
> Amazon Linux 2 sta per terminare il supporto. Per ulteriori informazioni, consulta Amazon Linux 2 FAQs.

- In questo modello, l'host bastion si trova in una sottorete privata senza un gateway NAT e un gateway Internet. Questo design isola l' EC2 istanza Amazon dalla rete Internet pubblica. Puoi aggiungere una configurazione di rete specifica che le consenta di comunicare con Internet. Per ulteriori informazioni, consulta Connect your virtual private cloud (VPC) ad altre reti nella documentazione di Amazon VPC. Allo stesso modo, seguendo il principio del privilegio minimo, l'host bastion non ha accesso ad altre risorse del tuo sistema a Account AWS meno che tu non conceda esplicitamente le autorizzazioni. Per ulteriori informazioni, consulta le politiche basate sulle risorse nella documentazione IAM.

Versioni del prodotto

- AWS CLI versione 2
- Terraform versione 1.3.9

Architecture

Stack tecnologico Target

- Un VPC con un'unica sottorete privata
- I seguenti endpoint VPC di interfaccia:
  - `amazonaws.<region>.ssm`— L'endpoint del servizio. AWS Systems Manager
  - `amazonaws.<region>.ec2messages`— Systems Manager utilizza questo endpoint per effettuare chiamate da SSM Agent al servizio Systems Manager.
  - `amazonaws.<region>.ssmmessages`— Session Manager utilizza questo endpoint per connettersi alla tua EC2 istanza Amazon tramite un canale dati sicuro.
- Un' EC2 istanza `t3.nano` Amazon che esegue Amazon Linux 2
- Ruolo e profilo dell'istanza IAM
- Gruppi di sicurezza Amazon VPC e regole dei gruppi di sicurezza per gli endpoint e l'istanza Amazon EC2

Architettura di Target

Il diagramma mostra il seguente processo:

1. L'utente assume un ruolo IAM con le autorizzazioni per eseguire le seguenti operazioni:
   - Autentica, autorizza e connettiti all'istanza Amazon EC2
   - Inizia una sessione con Session Manager
2. L'utente avvia una sessione SSH tramite Session Manager.
3. Session Manager autentica l'utente, verifica le autorizzazioni nelle politiche IAM associate, controlla le impostazioni di configurazione e invia un messaggio all'agente SSM per aprire una connessione bidirezionale.

4. L'utente invia la chiave pubblica SSH all'host bastion tramite i metadati di Amazon. EC2 Questa operazione deve essere eseguita prima di ogni connessione. La chiave pubblica SSH rimane disponibile per 60 secondi.
5. L'host bastion comunica con gli endpoint di interfaccia VPC per Systems Manager e Amazon. EC2
6. L'utente accede all'host bastion tramite Session Manager utilizzando un canale di comunicazione bidirezionale crittografato TLS 1.2.

Automazione e scalabilità

Sono disponibili le seguenti opzioni per automatizzare l'implementazione o scalare questa architettura:

- È possibile implementare l'architettura tramite una pipeline di integrazione e distribuzione continua (CI/CD).
- È possibile modificare il codice per cambiare il tipo di istanza del bastion host.
- Puoi modificare il codice per distribuire più host bastion. Nel `bastion-host/main.tf` file, nel blocco di `aws_instance` risorse, aggiungi il `count` meta-argomento. Per ulteriori informazioni, consulta la documentazione di Terraform.

Tools (Strumenti)

Servizi AWS

- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Systems Managerti aiuta a gestire le applicazioni e l'infrastruttura in esecuzione in. Cloud AWS Semplifica la gestione delle applicazioni e delle risorse, riduce i tempi di rilevamento e risoluzione dei problemi operativi e aiuta a gestire le AWS risorse in modo sicuro su larga scala. Questo modello utilizza Session Manager, una funzionalità di Systems Manager.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.

Altri strumenti

- HashiCorp Terraform è uno strumento Infrastructure as Code (IaC) che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud. Questo modello utilizza Terraform CLI.

Archivio di codice

Il codice per questo pattern è disponibile nell'host GitHub Access a bastion utilizzando Session Manager e Amazon EC2 Instance Connect.

Best practice

- Ti consigliamo di utilizzare strumenti automatici di scansione del codice per migliorare la sicurezza e la qualità del codice. Questo modello è stato scansionato utilizzando Checkov, uno strumento statico di analisi del codice per IaC. Come minimo, ti consigliamo di eseguire controlli di convalida e formattazione di base utilizzando i comandi e Terraform. `terraform validate` `terraform fmt -check -recursive`
- È buona norma aggiungere test automatici per IAc. Per ulteriori informazioni sui diversi approcci per testare il codice Terraform, consulta Testing HashiCorp Terraform (post sul blog Terraform).
- Durante la distribuzione, Terraform utilizza l' EC2 istanza Amazon sostitutiva ogni volta che viene rilevata una nuova versione dell'AMI Amazon Linux 2. Questo implementa la nuova versione del sistema operativo, incluse patch e aggiornamenti. Se la pianificazione della distribuzione non è frequente, ciò può rappresentare un rischio per la sicurezza perché l'istanza non dispone delle patch più recenti. È importante aggiornare e applicare frequentemente patch di sicurezza alle istanze Amazon EC2 distribuite. Per ulteriori informazioni, consulta Gestione degli aggiornamenti in Amazon EC2.
- Poiché questo modello è una dimostrazione di fattibilità, utilizza politiche AWS gestite, come`AmazonSSMManagedInstanceCore`. AWS le politiche gestite coprono casi d'uso comuni ma non concedono autorizzazioni con privilegi minimi. Se necessario per il tuo caso d'uso, ti consigliamo di creare policy personalizzate che concedano le autorizzazioni con il minimo privilegio per le risorse distribuite in questa architettura. Per ulteriori informazioni, consulta Introduzione alle politiche AWS gestite e passaggio alle autorizzazioni con privilegi minimi.
- Usa una password per proteggere l'accesso alle chiavi SSH e archivia le chiavi in un luogo sicuro.
- Configura la registrazione e il monitoraggio per l'host bastion. La registrazione e il monitoraggio sono parti importanti della manutenzione dei sistemi, sia dal punto di vista operativo che di sicurezza. Esistono diversi modi per monitorare le connessioni e le attività nel tuo bastion host. Per ulteriori informazioni, vedere i seguenti argomenti nella documentazione di Systems Manager:

- Monitoraggio AWS Systems Manager
- Registrazione e monitoraggio AWS Systems Manager
- Attività di controllo della sessione
- Attività di registrazione delle sessioni

Epiche

Implementa le risorse

Operazione	Description	Competenze richieste
Clona il repository del codice.	1. In un'interfaccia a riga di comando, modificate la directory di lavoro nella posizione in cui desiderate archiviare i file di esempio. 2. Inserire il seguente comando. `git clone https://github.com/aws-samples/secured-bastion-host-terraform.git`	DevOps ingegnere, sviluppatore
Inizializza la directory di lavoro di Terraform.	Questo passaggio è necessario solo per la prima implementazione. Se state ridistribuendo il pattern, passate al passaggio successivo. Nella directory principale del repository clonato, inserisci il seguente comando, dove: • `$S3_STATE_BUCKET` è il nome del bucket Amazon	DevOps ingegnere, sviluppatore, Terraform

Operazione	Description	Competenze richieste
	S3 che contiene lo stato Terraform • `$PATH_TO_STATE_FILE` è la chiave del file di stato Terraform, ad esempio `infra/bastion-host/tetfstate` • `$AWS_REGION` è la regione in cui viene distribuito il bucket Amazon S3 `terraform init \ -backend-config="bucket=$S3_STATE_BUCKET" \ -backend-config="key=$PATH_TO_STATE_FILE" \ -backend-config="region=$AWS_REGION` **Note** In alternativa, puoi aprire il file config.tf e, nella `terraform` sezione, fornire manualmente questi valori.	

Operazione	Description	Competenze richieste
Distribuisci le risorse.	1. Nella directory principal e del repository clonato, immettete il seguente comando. `terraform apply -var-file="dev.tfvars"` 2. Controlla l'elenco di tutte le modifiche che verranno applicate al tuo Account AWS, quindi conferma la distribuzione. 3. Attendi che tutte le risorse siano state distribuite.	DevOps ingegnere, sviluppat ore, Terraform

Configura l'ambiente locale

Operazione	Description	Competenze richieste
Configura la connessione SSH.	Aggiorna il file di configura zione SSH per consentire le connessioni SSH tramite Session Manager. Per istruzioni, consulta Consentir e le connessioni SSH per Session Manager. Ciò consente agli utenti autorizza ti di immettere un comando proxy che avvia una sessione di Session Manager e trasferis ce tutti i dati tramite una connessione bidirezionale.	DevOps ingegnere

Operazione	Description	Competenze richieste
Genera le chiavi SSH.	Immettere il seguente comando per generare una coppia di chiavi SSH pubblica e privata locale. Utilizzi questa key pair per connetterti all'host bastion. `ssh-keygen -t rsa -f my_key`	DevOps ingegnere, sviluppatore

Connect all'host bastion utilizzando Session Manager

Operazione	Description	Competenze richieste
Ottieni l'ID dell'istanza.	1. Per connetterti all'host bastion distribuito, è necessario l'ID dell'istanza Amazon EC2 . Effettua una delle seguenti operazioni per individuare l'ID: • Apri la EC2 console Amazon. Nel riquadro di navigazione, scegliere Instances (Istanze). Individua l'istanza bastion host. • Nel AWS CLI, inserisci il seguente comando. `aws ec2 describe-instances` Per filtrare i risultati , inserisci il seguente	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	comando, `$BASTION_HOST_TAG` dov'è il tag che hai assegnato al bastion host. Il valore predefinito per questo tag è `sandbox-dev-bastion-host` . `aws ec2 describe-instances \ --filters "Name=tag:Name,Values=$BASTION_HOST_TAG" \ --output text \ --query 'Reservations[*].Instances[*].InstanceId' \ --output text` 2. Copia l'ID dell' EC2 istanza Amazon. Utilizzerai questo ID in un secondo momento.	

Operazione	Description	Competenze richieste
Invia la chiave pubblica SSH.	**Note** In questa sezione, carichi la chiave pubblica nei metadati dell'istanza dell'host bastion. Dopo aver caricato la chiave, hai 60 secondi per avviare una connessione con l'host del bastion. Dopo 60 secondi, la chiave pubblica viene rimossa. Per ulteriori informazioni, consultate la sezione Risoluzione dei problemi di questo modello. Completa rapidamente i passaggi successivi per evitare che la chiave venga rimossa prima di connetterti al bastion host. 1. Invia la chiave SSH all'host bastion utilizzando Amazon Instance EC2 Connect. Inserisci il seguente comando, dove: • `$INSTANCE_ID` è l'ID dell' EC2 istanza Amazon	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	• `$PUBLIC_KEY_FILE` è il percorso del tuo file di chiave pubblica, ad esempio `my_key.pub` **Important** Assicurati di utilizzare la chiave pubblica e non la chiave privata. `aws ec2-instance-connect send-ssh-public-key \ --instance-id $INSTANCE_ID \ --instance-os-user ec2-user \ --ssh-public-key file://$PUBLIC_KEY_FILE` 2. Attendi di ricevere un messaggio che indica che la chiave è stata caricata correttamente. Passa immediatamente al passaggio successivo.	

Operazione	Description	Competenze richieste
Connect al bastion host.	1. Inserisci il seguente comando per connetter ti all'host bastion tramite Session Manager, dove: • `$PRIVATE_KEY_FILE` è il percorso della tua chiave privata, ad esempio `my_key` • `$INSTANCE_ID` è l'ID dell' EC2 istanza Amazon `ssh -i $PRIVATE_ KEY_FILE ec2-user@ $INSTANCE_ID` 2. Conferma la connessione inserendo`yes`. Si apre una connessione SSH utilizzan do Session Manager. **Note** Esistono altre opzioni per aprire una connessione SSH con l'host bastion. Per ulteriori informazi oni, consulta Approcci alternativi per stabilire una connessione SSH con l'host bastion nella sezione Informazioni	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	aggiuntive di questo modello.	

(Facoltativo) Pulizia

Operazione	Description	Competenze richieste
Rimuovi le risorse distribuite.	1. Per rimuovere tutte le risorse distribuite, esegui il comando seguente dalla directory principale del repository clonato. `terraform destroy -var-file="dev.tfvars"` 2. Conferma la rimozione delle risorse.	DevOps ingegnere, sviluppatore, Terraform

risoluzione dei problemi

Problema	Soluzione
`TargetNotConnected` errore durante il tentativo di connessione all'host bastion	1. Riavvia l'host bastion in base alle istruzioni in Riavvia l'istanza nella documentazione di Amazon. EC2 2. Dopo che l'istanza è stata riavviata con successo, invia nuovamente la chiave pubblica all'host bastion e ritenta la connessione.
`Permission denied` errore durante il tentativo di connessione all'host bastion	Dopo aver caricato la chiave pubblica sul bastion host, hai solo 60 secondi per avviare la

Problema	Soluzione
	connessione. Dopo 60 secondi, la chiave viene rimossa automaticamente e non puoi usarla per connetterti all'istanza. In tal caso, puoi ripetere il passaggio per inviare nuovamente la chiave all'istanza.

Risorse correlate

AWS documentazione

- AWS Systems Manager Session Manager(documentazione Systems Manager)
- Installa il plug-in Session Manager per AWS CLI (documentazione di Systems Manager)
- Consentire connessioni SSH per Session Manager (documentazione di Systems Manager)
- Informazioni sull'utilizzo di EC2 Instance Connect (EC2 documentazione Amazon)
- Connessione tramite EC2 Instance Connect (EC2 documentazione Amazon)
- Gestione delle identità e degli accessi per Amazon EC2 (EC2 documentazione Amazon)
- Utilizzo di un ruolo IAM per concedere autorizzazioni alle applicazioni in esecuzione su EC2 istanze Amazon (documentazione IAM)
- Le migliori pratiche di sicurezza in IAM (documentazione IAM)
- Controlla il traffico verso le risorse utilizzando gruppi di sicurezza (documentazione Amazon VPC)

Altre risorse

- Pagina web Terraform Developer
- Comando: validate (documentazione Terraform)
- Comando: fmt (documentazione Terraform)
- Testare HashiCorp Terraform (post sul blog) HashiCorp
- Pagina web di Checkov

Informazioni aggiuntive

Approcci alternativi per stabilire una connessione SSH con l'host bastion

Inoltro alla porta

È possibile utilizzare l'`-D 8888`opzione per aprire una connessione SSH con port forwarding dinamico. Per ulteriori informazioni, consulta le istruzioni su explainshell.com. Di seguito è riportato un esempio di comando per aprire una connessione SSH utilizzando il port forwarding.

```
ssh -i $PRIVATE_KEY_FILE -D 8888 ec2-user@$INSTANCE_ID
```

Questo tipo di connessione apre un proxy SOCKS in grado di inoltrare il traffico dal browser locale attraverso l'host bastion. Se usi Linux o macOS, per vedere tutte le opzioni, inserisci. `man ssh` Viene visualizzato il manuale di riferimento SSH.

Utilizzando lo script fornito

Invece di eseguire manualmente i passaggi descritti in Connect to the bastion host utilizzando Session Manager nella sezione Epics, puoi utilizzare lo script connect.sh incluso nel repository del codice. Questo script genera la coppia di chiavi SSH, invia la chiave pubblica all' EC2 istanza Amazon e avvia una connessione con l'host bastion. Quando esegui lo script, passi il tag e il nome della chiave come argomenti. Di seguito è riportato un esempio del comando per eseguire lo script.

```
./connect.sh sandbox-dev-bastion-host my_key
```

Centralizza la risoluzione DNS utilizzando AWS Managed Microsoft AD e localmente Microsoft Active Directory

Brian Westmoreland, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni per centralizzare la risoluzione DNS all'interno di un ambiente AWS multi-account utilizzando sia AWS Directory Service for Microsoft Active Directory () che Amazon Route AWS Managed Microsoft AD 53. In questo modello, lo spazio dei nomi AWS DNS è un sottodominio dello spazio dei nomi DNS locale. Questo modello fornisce anche indicazioni su come configurare i server DNS locali per inoltrare le query a AWS quando la soluzione DNS locale utilizza Microsoft Active Directory.

Prerequisiti e limitazioni

Prerequisiti

- Un ambiente AWS multi-account configurato utilizzando. AWS Organizations
- Connettività di rete stabilita tra Account AWS.
- Connettività di rete stabilita tra AWS e l'ambiente locale (utilizzando AWS Direct Connect o qualsiasi tipo di connessione VPN).
- AWS Command Line Interface (AWS CLI) configurato su una workstation locale.
- AWS Resource Access Manager (AWS RAM) utilizzato per condividere le regole della Route 53 tra account. Pertanto, la condivisione deve essere abilitata all'interno AWS Organizations dell'ambiente, come descritto nella sezione Epics.

Limitazioni

- AWS Managed Microsoft AD La Standard Edition ha un limite di 5 condivisioni.
- AWS Managed Microsoft AD Enterprise Edition ha un limite di 125 condivisioni.
- La soluzione in questo modello si limita alla Regioni AWS condivisione del supporto AWS RAM.

Versioni del prodotto

- Microsoft Active Directory in esecuzione su Windows Server 2008, 2012, 2012 R2 o 2016.

Architecture

Architettura Target

In questo design, AWS Managed Microsoft AD viene installato nei servizi condivisi. Account AWS Sebbene non sia un requisito, questo modello presuppone questa configurazione. Se configuri AWS Managed Microsoft AD in modo diverso Account AWS, potresti dover modificare di conseguenza i passaggi nella sezione Epics.

Questo design utilizza i Resolver Route 53 per supportare la risoluzione dei nomi tramite l'uso delle regole Route 53. Se la soluzione DNS locale utilizza Microsoft DNS, la creazione di una regola di inoltro condizionale per il namespace AWS `aws.company.com` (), che è un sottodominio dello spazio dei nomi DNS dell'azienda (), non è semplice. `company.com` Se si tenta di creare un server d'inoltro condizionale tradizionale, si verificherà un errore. Questo perché Microsoft Active Directory è già considerato autorevole per qualsiasi sottodominio di. `company.com` Per aggirare questo errore, devi prima creare una `aws.company.com` delega per delegare l'autorità di quel namespace. È quindi possibile creare il server d'inoltro condizionale.

Il cloud privato virtuale (VPC) per ogni account spoke può avere il proprio spazio dei nomi DNS univoco basato sullo spazio dei nomi root. AWS In questo design, ogni account spoke aggiunge un'abbreviazione del nome dell'account allo spazio dei nomi AWS di base. Dopo aver creato le zone ospitate private nell'account spoke, le zone vengono associate al VPC locale nell'account spoke e al VPC nell'account di rete centrale. AWS Ciò consente all'account di AWS rete centrale di rispondere alle domande DNS relative agli account spoke. In questo modo, sia Route 53 che Route 53 AWS Managed Microsoft AD collaboriamo per condividere la responsabilità della gestione del AWS namespace (). `aws.company.com`

Automazione e scalabilità

Questo design utilizza gli endpoint Route 53 Resolver per scalare le query DNS tra AWS l'ambiente locale e l'ambiente locale. Ogni endpoint Route 53 Resolver comprende più interfacce di rete elastiche (distribuite su più zone di disponibilità) e ogni interfaccia di rete può gestire fino a 10.000 query al secondo. Route 53 Resolver supporta fino a 6 indirizzi IP per endpoint, quindi complessivamente questo design supporta fino a 60.000 query DNS al secondo distribuite su più zone di disponibilità per un'elevata disponibilità.

Inoltre, questo modello tiene automaticamente conto delle crescite future interne AWS. Le regole di inoltro DNS configurate in locale non devono essere modificate per supportare le nuove zone ospitate private VPCs e le relative zone ospitate private associate che vengono aggiunte. AWS

Tools (Strumenti)

Servizi AWS

- AWS Directory Service for Microsoft Active Directoryconsente ai carichi di lavoro e alle risorse compatibili con le directory di AWS utilizzare Microsoft Active Directory in. Cloud AWS
- AWS Organizationsè un servizio di gestione degli account che consente di consolidare più account Account AWS in un'organizzazione creata e gestita centralmente.
- AWS Resource Access Manager (AWS RAM) ti aiuta a condividere in modo sicuro le tue risorse Account AWS per ridurre il sovraccarico operativo e fornire visibilità e verificabilità.
- Amazon Route 53 è un servizio Web DNS altamente scalabile e disponibile.

Strumenti

- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando. In questo modello, AWS CLI viene utilizzato per configurare le autorizzazioni di Route 53.

Epiche

Crea e condividi una directory AWS Managed Microsoft AD

Operazione	Description	Competenze richieste
Implementazione. AWS Managed Microsoft AD	1. Crea e configura una nuova directory. Per i passaggi dettagliati, consulta Creating your AWS Managed Microsoft AD nella AWS Directory Service Administration Guide. 2. Registra gli indirizzi IP dei controller di AWS Managed Microsoft AD dominio. A	Amministratore AWS

Operazione	Description	Competenze richieste
	questi verrà fatto riferimento in un passaggio successivo.	
Condividi la directory.	Dopo aver creato la directory , condividila con altri Account AWS membri dell' AWS organizzazione. Per istruzioni, consulta Condividere la directory nella Guida all'AWS Directory Service amministrazione. **Note** AWS Managed Microsoft AD La Standard Edition ha un limite di 5 condivisioni. Enterprise Edition ha un limite di 125 condivisioni.	Amministratore AWS

Configura Route 53

Operazione	Description	Competenze richieste
Crea risolutori Route 53.	I resolver Route 53 facilitano la risoluzione delle query DNS tra e il data center locale. AWS 1. Installa i risolutori Route 53 seguendo le istruzioni contenute nella Guida per gli sviluppatori di Route 53.	Amministratore AWS

Operazione	Description	Competenze richieste
	2. Configura i Resolver Route 53 in sottoreti private in almeno due zone di disponibilità all'interno dell' AWS account di rete centrale VPC per un'elevata disponibilità. **Note** Sebbene l'utilizzo dell'account di AWS rete centrale VPC non sia un requisito, i passaggi rimanenti presuppongono questa configurazione.	

Operazione	Description	Competenze richieste

Crea regole Route 53.	Il tuo caso d'uso specifico potrebbe richiedere un gran numero di regole Route 53, ma dovrai configurare le seguenti regole come base:
	• Una regola in uscita per lo spazio dei nomi locale (`company.com`) che utilizza l'account di rete centrale Route 53 Resolvers in uscita. Gli indirizzi IP di destinazione sono i server DNS locali.
	  • Associa questa regola all'account di rete centrale VPC.
	• Una regola in uscita per il AWS namespace (`aws.company.com`) utilizzando l'account di rete centrale Route 53 Resolvers in uscita. Gli indirizzi IP di destinazione sono gli indirizzi IP di Route 53 Resolver in entrata dell'account di rete centrale.
	  • Non associare questa regola all'account di AWS rete centrale VPC (che ospita i Resolver Route 53).
	• Una seconda regola in uscita per il AWS
		Amministratore AWS

Operazione	Description	Competenze richieste
	namespace (`aws.company.com`) che punta ai controller di AWS Managed Microsoft AD dominio (usa quella dell'epic precedente). IPs • Associate questa regola all'account di AWS rete centrale VPC (che ospita i Resolver Route 53). • Non condividere o associare questa regola ad altri. Account AWS Per ulteriori informazioni, consulta Managing forwarding rules nella Route 53 Developer Guide.	

Operazione	Description	Competenze richieste
Configura un profilo Route 53.	Un profilo Route 53 viene utilizzato per condividere le regole con gli account Spoke. 1. Crea un nuovo profilo Route 53 nell'account di rete centrale seguendo le istruzioni nella Route 53 Developer Guide. 2. Aggiungi la regola per lo spazio dei nomi locale (`company.com`) al profilo. 3. Aggiungi la prima regola per il AWS namespace (`aws.company.com`), che ha come target gli indirizzi IP dei resolver in ingresso della Route 53, al profilo. 4. Condividi il profilo Route 53 con l'organizzazione. AWS 5. Accetta la condivisione delle risorse del profilo Route 53 in ogni account spoke. 6. Associa il profilo Route 53 a ciascun VPC dell'account spoke.	Amministratore AWS

Configura il DNS di Active Directory locale

Operazione	Description	Competenze richieste
Crea la delega.	Utilizza lo snap-in Microsoft DNS (`dnsmgmt.msc`) per creare una nuova delega per lo spazio dei `company.com` nomi all'interno di Active Directory. Il nome del dominio delegato deve essere. `aws` Ciò costituisce il nome di dominio completo (FQDN) della delegazione. `aws.company.com` Utilizza gli indirizzi IP dei controller di AWS Managed Microsoft AD dominio per i valori IP del name server e usali `server.aws.company.com` per il nome. (Questa delega serve solo per la ridondanza, poiché per questo spazio dei nomi verrà creato un server d'inoltro condizionale che ha la precedenza sulla delega.)	Active Directory
Crea lo spedizioniere condizionale.	Usa lo snap-in Microsoft DNS (`dnsmgmt.msc`) per creare un nuovo server d'inoltro condizionale per. `aws.company.com` Utilizza gli indirizzi IP dei Resolver Route 53 AWS in ingresso nel DNS centrale per la destinazione del server	Active Directory

Operazione	Description	Competenze richieste
	d'inoltro condizionale. Account AWS	

Crea zone private ospitate su Route 53 per spoke Account AWS

Operazione	Description	Competenze richieste
Crea le zone ospitate private di Route 53.	Crea una zona ospitata privata sulla Route 53 in ogni account spoke. Associa questa zona ospitata privata al VPC dell'account spoke. Per i passaggi dettagliati, consulta Creazione di una zona ospitata privata nella Route 53 Developer Guide.	Amministratore AWS
Crea autorizzazioni.	Utilizza AWS CLI per creare un'autorizzazione per l'account di AWS rete centrale VPC. Esegui questo comando dal contesto di ogni spoke Account AWS: `aws route53 create-vpc-association-authorization --hosted-zone-id <hosted-zone-id> \ --vpc VPCRegion=<region>,VPCId=<vpc-id>` dove:	Amministratore AWS

Operazione	Description	Competenze richieste
	• <hosted-zone-id> è la zona ospitata privata di Route 53 nell'account spoke. • <region>e <vpc-id> sono l' Regione AWS ID VPC dell'account di AWS rete centrale VPC.	
Creare associazioni.	Crea l'associazione di zona ospitata privata Route 53 per l'account di AWS rete centrale VPC utilizzando. AWS CLI Esegui questo comando dal contesto dell'account di AWS rete centrale: `aws route53 associate-vpc-with-hosted-zone --hosted-zone-id <hosted-zone-id> \ --vpc VPCRegion=<region>,VPCId=<vpc-id>` dove: • <hosted-zone-id> è la zona ospitata privata di Route 53 nell'account spoke. • <region>e <vpc-id> sono l'ID Regione AWS e il VPC dell'account di AWS rete centrale.	Amministratore AWS

Risorse correlate

- Semplifica la gestione DNS in un ambiente multi-account con Route 53 Resolver (post sul blog)AWS
- Creando la tua (documentazione) AWS Managed Microsoft ADAWS Directory Service
- Condivisione di una AWS Managed Microsoft AD directory (AWS Directory Service documentazione)
- Che cos'è Amazon Route 53 Resolver? (documentazione Amazon Route 53)
- Creazione di una zona ospitata privata (documentazione Amazon Route 53)
- Cosa sono i profili Amazon Route 53? (documentazione Amazon Route 53)

Centralizza il monitoraggio utilizzando Amazon CloudWatch Observability Access Manager

Anand Krishna Varanasi, JAGDISH KOMAKULA, Ashish Kumar, Jimmy Morgan, Sarat Chandra Pothula, Vivek Thangamuthu e Balaji Vedagiri, Amazon Web Services

Riepilogo

L'osservabilità è fondamentale per il monitoraggio, la comprensione e la risoluzione dei problemi delle applicazioni. Le applicazioni che si estendono su più account, come nelle AWS Control Tower nostre implementazioni di landing zone, generano un gran numero di log e dati di traccia. Per risolvere rapidamente i problemi o comprendere l'analisi degli utenti o l'analisi aziendale, è necessaria una piattaforma di osservabilità comune per tutti gli account. Amazon CloudWatch Observability Access Manager ti consente di accedere e controllare più log di account da una posizione centrale.

Puoi utilizzare Observability Access Manager per visualizzare e gestire i log dei dati di osservabilità generati dagli account di origine. Gli account di origine sono singoli Account AWS che generano dati di osservabilità per le proprie risorse. I dati di osservabilità vengono condivisi tra gli account di origine e gli account di monitoraggio. I dati di osservabilità condivisi possono includere metriche in Amazon CloudWatch, log in Amazon CloudWatch Logs e tracce in. AWS X-Ray Per ulteriori informazioni, consulta la documentazione di Observability Access Manager.

Questo modello è destinato agli utenti che dispongono di applicazioni o infrastrutture eseguite su più sistemi Account AWS e necessitano di un posto comune in cui visualizzare i log. Spiega come configurare Observability Access Manager utilizzando Terraform, per monitorare lo stato e l'integrità di queste applicazioni o infrastrutture. È possibile installare questa soluzione in diversi modi:

- Come modulo Terraform autonomo da configurare manualmente
- Utilizzando una pipeline di integrazione e distribuzione continua (CI/CD)
- Integrandosi con altre soluzioni come AWS Control Tower Account Factory for Terraform (AFT)

Le istruzioni nella sezione Epics riguardano l'implementazione manuale. Per i passaggi di installazione di AFT, consulta il file README per il repository GitHub Observability Access Manager.

Prerequisiti e limitazioni

Prerequisiti

- Terraform è installato o referenziato nel sistema o in pipeline automatizzate. (Ti consigliamo di utilizzare la versione più recente.)
- Un account che puoi utilizzare come account di monitoraggio centralizzato. Altri account creano collegamenti all'account di monitoraggio centrale per visualizzare i registri.
- (Facoltativo) Un archivio di codice sorgente come GitHub Atlassian Bitbucket o un sistema simile. AWS CodeCommit Un archivio di codice sorgente non è necessario se utilizzi pipeline automatizzate. CI/CD
- (Facoltativo) Autorizzazioni per creare richieste pull (PRs) per la revisione e la collaborazione del codice in. GitHub

Limitazioni

Observability Access Manager ha le seguenti quote di servizio, che non possono essere modificate. Considerate queste quote prima di implementare questa funzionalità. Per ulteriori informazioni, consulta le quote CloudWatch di servizio nella documentazione. CloudWatch

- Collegamenti agli account di origine: puoi collegare ciascun account di origine a un massimo di cinque account di monitoraggio.
- Lavandini: puoi creare più sink per un account, ma Regione AWS è consentito un solo sink per account.

Inoltre:

- I sink e i link devono essere creati nello stesso ambiente Regione AWS; non possono essere interregionali.

Monitoraggio interregionale e interaccount

Per il monitoraggio interregionale e tra più account, puoi scegliere una di queste opzioni:

- Crea CloudWatch dashboard per più account e più regioni per allarmi e metriche. Questa opzione non supporta log e tracce.
- Implementa la registrazione centralizzata utilizzando Amazon OpenSearch Service.
- Crea un sink per regione da tutti gli account tenant, invia i parametri a un account di monitoraggio centralizzato (come descritto in questo schema), quindi utilizza i flussi di CloudWatch metrici per

inviare i dati a una destinazione esterna comune o a prodotti di monitoraggio di terze parti come Datadog, Dynatrace, Sumo Logic, Splunk o New Relic.

Architecture

Componenti

CloudWatch Observability Access Manager è costituito da due componenti principali che consentono l'osservabilità tra account:

- Un sink offre agli account di origine la possibilità di inviare dati di osservabilità all'account di monitoraggio centrale. Un sink fornisce fondamentalmente una giunzione gateway a cui gli account di origine possono connettersi. Può esserci solo un gateway o una connessione sink e più account possono connettersi ad esso.
- Ogni account di origine ha un collegamento alla giunzione del sink gateway e i dati di osservabilità vengono inviati tramite questo collegamento. È necessario creare un sink prima di creare collegamenti da ciascun account di origine.

Architettura

Il diagramma seguente illustra Observability Access Manager e i suoi componenti.

Tools (Strumenti)

Servizi AWS

- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue AWS risorse e delle applicazioni su cui esegui AWS in tempo reale.
- AWS Organizationsè un servizio di gestione degli account che ti aiuta a consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.

Strumenti

- Terraform è uno strumento di infrastruttura come codice (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.

- AWS Control Tower Account Factory for Terraform (AFT) imposta una pipeline Terraform per aiutarti a fornire e personalizzare gli account. AWS Control Tower Facoltativamente, puoi utilizzare AFT per configurare Observability Access Manager su larga scala su più account.

Archivio di codice

Il codice per questo modello è disponibile nel repository di GitHub Observability Access Manager.

Best practice

- Negli AWS Control Tower ambienti, contrassegna l'account di registrazione come account di monitoraggio centrale (sink).
- Se hai più organizzazioni con più account AWS Organizations, ti consigliamo di includere le organizzazioni anziché i singoli account nella politica di configurazione. Se disponi di un numero limitato di account o se gli account non fanno parte di un'organizzazione nella politica di configurazione del sink, potresti decidere di includere invece account individuali.

Epiche

Configura il modulo lavandino

Operazione	Description	Competenze richieste
Clonare il repository.	Clonare il repository di GitHub Observability Access Manager: `git clone https://github.com/aws-samples/cloudwatch-observability-access-manager-terraform`	AWS DevOps, amministratore cloud, amministratore AWS
Specificare i valori delle proprietà per il modulo sink.	Nel `main.tf` file (nella `deployments/aft-account-customizations/LOGGING/terraform/` cartella del repositor	AWS DevOps, amministratore cloud, amministratore AWS

Operazione	Description	Competenze richieste
	y), specificate i valori per le seguenti proprietà: • `sink_name` : Il nome del CloudWatch lavello. • `allowed_oam_resource_types` : Observability Access Manager attualmente supporta CloudWatch metriche, gruppi di log e AWS X-Ray tracce. • `allowed_source_accounts` : Gli account di origine a cui è consentito inviare registri all'account central CloudWatch sink. • `allowed_source_organizations` : le AWS Control Tower organizzazioni di origine a cui è consentito inviare registri all'account Central CloudWatch Sink. Per ulteriori informazioni, AWS::Oam::Sinkconsulta la CloudFormation documentazione.	

Operazione	Description	Competenze richieste

Installa il modulo sink.	Esporta le credenziali dell'account Account AWS che hai selezionato come account di monitoraggio e installa il modulo sink di Observability Access Manager:
	`Terraform Init`
	`Terrafom Plan`
	`Terraform Apply`
		AWS DevOps, amministratore cloud, amministratore AWS

Configura il modulo di collegamento

Operazione	Description	Competenze richieste

Specificate i valori delle proprietà per il modulo di collegamento.	Nel `main.tf` file (nella `deployments/aft-account-customizations/LOGGING/terraform/` cartella del repository), specificate i valori per le seguenti proprietà:
	• `account_label` : Utilizzate uno dei seguenti valori:
	  • `$AccountName` : il nome dell'account.
	  • `$AccountEmail` : un indirizzo e-mail unico a livello globale, che include il dominio e-mail (ad esempio,`hello@example.com`)
		AWS DevOps, amministratore cloud, architetto cloud

Operazione	Description	Competenze richieste
	• `$AccountEmailNoDomain` : un indirizzo e-mail senza il nome di dominio. • `allowed_oam_resource_types` : Observability Access Manager attualmente supporta CloudWatch metriche, gruppi di log e AWS X-Ray tracce. Per ulteriori informazioni, consulta la AWS::Oam::Link CloudFormation documentazione.	

Installa il modulo di collegamento per i singoli account.	Esporta le credenziali dei singoli account e installa il modulo di collegamento Observability Access Manager:
	
	`Terraform Plan`
	`Terraform Apply`
	
	È possibile configurare il modulo di collegamento individualmente per ciascun account o utilizzare AFT per installare automaticamente questo modulo su un gran numero di account.
		AWS DevOps, amministratore cloud, architetto cloud

Approva le connessioni sink-to-link

Operazione	Description	Competenze richieste
Controlla il messaggio di stato.	1. Accedi all'account di monitoraggio. 2. Apri la CloudWatch console. 3. Nel riquadro di navigazione a sinistra scegliere Impostazioni. Sulla destra, dovresti vedere il messaggio di stato Monitoraggio dell'account abilitato con un segno di spunta verde. Ciò significa che l'account di monitoraggio dispone di un sink di Observability Access Manager a cui si collegheranno i collegamenti di altri account.	
Approva le connessioni. link-to-sink	1. Scegli l'opzione Risorse per collegare gli account sotto il messaggio di stato. Le informazioni confermano che si tratta dell'account di monitoraggio, elencano i dati condivisi dagli account di origine del tenant (Logs, Metrics, Traces) e mostrano l'etichetta dell'account come $. AccountName Questa schermata offre due opzioni per collegare gli	AWS DevOps, amministratore cloud, architetto cloud

Operazione	Description	Competenze richieste
	account tenant all'account di monitoraggio: approvazione a livello di organizzazione o approvazione a livello di account. Per ogni opzione, puoi scegliere di scaricare un CloudFormation modello per l'approvazione o approvare ogni account singolarmente. 2. Per semplicità, scegli Qualsiasi account da approvare a ogni livello di account. Questa opzione fornisce un link di approvazione per l'account. 3. Scegli Copia URL per copiare il link. 4. Accedi a ciascun account di origine. 5. In una finestra del browser, incolla il link e scegli Approva link connect to sink. 6. Ripeti l'operazione per altri account di origine. Per ulteriori informazioni, consulta Collegare gli account di monitoraggio agli account di origine nella CloudWatch documentazione.	

Verifica i dati di osservabilità tra account

Operazione	Description	Competenze richieste
Visualizza i dati relativi a più account.	1. Accedi all'account di monitoraggio centrale. 2. Apri la CloudWatch console. 3. Nel riquadro di navigazione a sinistra, scegli le opzioni per visualizzare i log, le metriche e le tracce di più account.	AWS DevOps, amministratore cloud, architetto cloud

(Facoltativo) Consenti agli account di origine di considerare attendibili gli account di monitoraggio

Operazione	Description	Competenze richieste
Visualizza metriche, dashboard, registri, widget e allarmi di altri account.	Come funzionalità aggiuntiva, puoi condividere CloudWatch metriche, dashboard, registri, widget e allarmi con altri account. Ogni account utilizza un ruolo IAM chiamato CloudWatch- CrossAccountSharingRole per accedere a questi dati. Gli account di origine che hanno un rapporto di fiducia con l'account di monitoraggio centrale possono assumere questo ruolo e visualizzare i dati dell'account di monitoraggio.	AWS DevOps, amministratore cloud, architetto cloud

Operazione	Description	Competenze richieste
	CloudWatch fornisce uno CloudFormation script di esempio per creare il ruolo. Scegli Gestisci il ruolo in IAM ed esegui questo script negli account in cui desideri visualizzare i dati. `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "AWS": [ "arn:aws:iam::XXXXXXXXXX:root", "arn:aws:iam::XXXXXXXXXX:root", "arn:aws:iam::XXXXXXXXXX:root", "arn:aws:iam::XXXXXXXXXX:root" ] }, "Action": "sts:AssumeRole" } ] }`	

Operazione	Description	Competenze richieste
	Per ulteriori informazioni, consulta Attivazione della funzionalità tra account CloudWatch nella CloudWatch documentazione.	

(Facoltativo) Visualizza più account Interregionali dall'account di monitoraggio

Operazione	Description	Competenze richieste
Configura l'accesso su più account e più regioni.	Nell'account di monitoraggio centralizzato, puoi aggiunger e facoltativamente un selettore di account per passare facilmente da un account all'altro e visualizzarne i dati senza dover autenticarti. 1. Accedi all'account di monitoraggio centralizzato. 2. Apri la CloudWatch console. 3. Nel riquadro di navigazio ne a sinistra scegliere Impostazioni. 4. Nella sezione Visualizza più account interregionali, scegli Configura. 5. Scegli Abilita, quindi seleziona la casella di controllo Mostra il selettore nella console. 6. Scegli una di queste opzioni:	AWS DevOps, amministratore cloud, architetto cloud

Operazione	Description	Competenze richieste
	• Inserimento dell'ID dell'account: questa opzione richiede di inserire manualmente l'ID dell'account ogni volta che desideri modificare gli account per visualizz are i dati dei diversi account. • Selettore di account AWS Organization: se hai effettuato l'integra zione CloudWatch con AWS Organizations, questa opzione fornisce un selettore a discesa con un elenco completo degli account dell'orga nizzazione. • Selettore di account personalizzato: questa opzione consente di inserire manualmente un elenco di account IDs per compilare il selettore. 7. Scegli Save changes (Salva modifiche). Per ulteriori informazioni, consulta Console interregi onale CloudWatch tra account nella documentazione. CloudWatch	

Risorse correlate

- CloudWatch osservabilità tra account (documentazione Amazon CloudWatch)
- Riferimento all'API Amazon CloudWatch Observability Access Manager (CloudWatch documentazione Amazon)
- Risorsa: aws_oam_sink (documentazione Terraform)
- Fonte dati: aws_oam_link (documentazione Terraform)
- CloudWatchObservabilityAccessManagerAWS (Documentazione Boto3)

Verifica la presenza di tag obbligatori EC2 nelle istanze al momento del lancio

Susanne Kangnoh e Archit Mathur, Amazon Web Services

Riepilogo

Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud Amazon Web Services (AWS). L'utilizzo di Amazon EC2 elimina la necessità di investire in hardware in anticipo, in modo da poter sviluppare e distribuire applicazioni più velocemente.

Puoi utilizzare i tag per classificare le tue risorse AWS in diversi modi. EC2 il tagging delle istanze è utile quando hai molte risorse nel tuo account e desideri identificare rapidamente una risorsa specifica in base ai tag. Puoi assegnare metadati personalizzati alle tue EC2 istanze utilizzando i tag. Un tag è costituito da una chiave e un valore definiti dall'utente. Ti consigliamo di creare un set coerente di tag per soddisfare i requisiti della tua organizzazione.

Questo modello fornisce un CloudFormation modello AWS per aiutarti a monitorare EC2 le istanze per tag specifici. Il modello crea un evento Amazon CloudWatch Events che controlla l'AWS CloudTrail TagResourceo UntagResourcegli eventi, per rilevare l'etichettatura o la rimozione di nuove EC2 istanze. Se manca un tag predefinito, richiama una funzione AWS Lambda, che invia un messaggio di violazione a un indirizzo e-mail fornito da te, utilizzando Amazon Simple Notification Service (Amazon SNS).

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) per caricare il codice Lambda fornito.
- Un indirizzo e-mail a cui desideri ricevere notifiche di violazione.

Limitazioni

- Questa soluzione supporta i CloudTrail TagResourcenostri UntagResourceeventi. Non crea notifiche per altri eventi.
- Questa soluzione verifica solo le chiavi dei tag. Non monitora i valori chiave.

Architecture

Architettura del workflow

Automazione e scalabilità

- Puoi utilizzare il CloudFormation modello AWS più volte per diverse regioni e account AWS. Devi eseguire il modello solo una volta in ogni regione o account.

Tools (Strumenti)

Servizi AWS

- Amazon EC2 — Amazon Elastic Compute Cloud (Amazon EC2) è un servizio Web che fornisce capacità di elaborazione sicura e ridimensionabile nel cloud. È progettato per semplificare il cloud computing su scala web per gli sviluppatori.
- AWS CloudTrail: CloudTrail è un servizio AWS che ti aiuta con la governance, la conformità e il controllo operativo e dei rischi del tuo account AWS. Le azioni intraprese da un utente, un ruolo o un servizio AWS vengono registrate come eventi in CloudTrail.
- Amazon CloudWatch Events: Amazon CloudWatch Events offre un flusso quasi in tempo reale di eventi di sistema che descrivono i cambiamenti nelle risorse AWS. CloudWatch Events viene a conoscenza dei cambiamenti operativi man mano che si verificano e intraprende le azioni correttive necessarie, inviando messaggi per rispondere all'ambiente, attivando funzioni, apportando modifiche e acquisendo informazioni sullo stato.
- AWS Lambda — Lambda è un servizio di elaborazione che supporta l'esecuzione di codice senza dover fornire o gestire server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti altamente scalabile che può essere utilizzato per un'ampia gamma di soluzioni di storage, tra cui siti Web, applicazioni mobili, backup e data lake.
- Amazon SNS — Amazon Simple Notification Service (Amazon SNS) è un servizio Web che consente ad applicazioni, utenti finali e dispositivi di inviare e ricevere istantaneamente notifiche dal cloud.

Codice

Questo modello include un allegato con due file:

- `index.zip`è un file compresso che include il codice Lambda per questo modello.
- `ec2-require-tags.yaml`è un CloudFormation modello che distribuisce il codice Lambda.

Consulta la sezione Epics per informazioni su come usare questi file.

Epiche

Implementa il codice Lambda

Operazione	Description	Competenze richieste
Carica il codice in un bucket S3.	Crea un nuovo bucket S3 o usa un bucket S3 esistente per caricare il file allegato `index.zip` (codice Lambda). Questo bucket deve trovarsi nella stessa regione AWS delle risorse (EC2 istanze) che desideri monitorare.	Architetto del cloud
Implementa il CloudFormation modello.	Apri la console Cloudformation nella stessa regione AWS del bucket S3 e distribuisci il `ec2-require-tags.yaml` file fornito nell'allegato. Nella prossima epopea, fornisci i valori per i parametri del modello.	Architetto del cloud

Completa i parametri nel CloudFormation modello

Operazione	Description	Competenze richieste
Fornisci il nome del bucket S3.	Inserisci il nome del bucket S3 che hai creato o seleziona to nella prima epic. Questo bucket S3 contiene il file.zip per il codice Lambda e deve trovarsi nella stessa regione AWS del CloudFormation modello e delle EC2 istanze che desideri monitorare.	Architetto del cloud
Fornisci la chiave S3.	Fornisci la posizione del file.zip del codice Lambda nel tuo bucket S3, senza barre iniziali (ad esempio o). `index.zip` `controls/index.zip`	Architetto del cloud
Fornisci un indirizzo email.	Fornisci un indirizzo email attivo a cui desideri ricevere le notifiche di violazione.	Architetto del cloud
Definisci un livello di registraz ione.	Specificare il livello di registraz ione e la verbosità. `Info`indica messaggi informativi dettaglia ti sullo stato di avanzamen to dell'applicazione e deve essere utilizzato solo per il debug. `Error`indica eventi di errore che potrebbero comunque consentire all'appli cazione di continuare a funzionare. `Warning`indica	Architetto del cloud

Operazione	Description	Competenze richieste
	situazioni potenzialmente dannose.	
Inserisci le chiavi dei tag richieste.	Inserisci le chiavi dei tag che desideri controllare. Se desideri specificare più chiavi, separale con virgole, senza spazi. (Ad esempio, `ApplicationId,CreatedBy,Environment,Organization` cerca quattro chiavi). L'evento CloudWatch Events cerca queste chiavi di tag e invia una notifica se non vengono trovate.	Architetto del cloud

Confermare la sottoscrizione

Operazione	Description	Competenze richieste
Conferma l'iscrizione via e-mail.	Quando il CloudFormation modello viene distribuito correttamente, invia un messaggio e-mail di sottoscrizione all'indirizzo e-mail fornito. Per ricevere notifiche, devi confermare questa sottoscrizione e-mail.	Architetto del cloud

Risorse correlate

- Creazione di un bucket (documentazione Amazon S3)
- Caricamento di oggetti (documentazione Amazon S3)

- Etichetta le tue EC2 risorse Amazon (EC2 documentazione Amazon)
- Creazione di una regola CloudWatch Events che si attiva su una chiamata API AWS utilizzando AWS CloudTrail (CloudWatch documentazione Amazon)

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Pulisci le risorse di AWS Account Factory for Terraform (AFT) in modo sicuro dopo la perdita dei file di stato

Gokendra Malviya, Amazon Web Services

Riepilogo

Quando utilizzi AWS Account Factory for Terraform (AFT) per gestire il tuo AWS Control Tower ambiente, AFT genera un file di stato Terraform per tenere traccia dello stato e della configurazione delle risorse create da Terraform. La perdita del file di stato Terraform può creare sfide significative per la gestione e la pulizia delle risorse. Questo modello fornisce un approccio sistematico per identificare e rimuovere in sicurezza le risorse relative ad AFT mantenendo l'integrità dell'ambiente. AWS Control Tower

Il processo è progettato per garantire la corretta rimozione di tutti i componenti AFT, anche senza il riferimento originale al file di stato. Questo processo fornisce un percorso chiaro per ristabilire e riconfigurare con successo AFT nell'ambiente, per contribuire a ridurre al minimo le interruzioni delle operazioni. AWS Control Tower

Per ulteriori informazioni su AFT, consultate la documentazione.AWS Control Tower

Prerequisiti e limitazioni

Prerequisiti

- Una conoscenza approfondita dell'architettura AFT.
- Accesso da amministratore ai seguenti account:
  - Account di gestione AFT
  - AWS Control Tower Account di gestione
  - Account Log Archive
  - Account di audit
- Verifica che nessuna politica di controllo del servizio (SCPs) contenga restrizioni o limitazioni che bloccherebbero l'eliminazione delle risorse relative ad AFT.

Limitazioni

- Questo processo può ripulire le risorse in modo efficace, ma non può recuperare i file di stato perduti e alcune risorse potrebbero richiedere l'identificazione manuale.

- La durata del processo di pulizia dipende dalla complessità dell'ambiente e potrebbe richiedere diverse ore.
- Questo modello è stato testato con la versione AFT 1.12.2 ed elimina le seguenti risorse. Se utilizzate una versione diversa di AFT, potrebbe essere necessario eliminare risorse aggiuntive.

Nome del servizio	Numero di risorse
AWS CodeBuild	6
AWS CodeCommit	4
AWS CodePipeline	4
Amazon DynamoDB	5
Amazon Elastic Compute Cloud (Amazon EC2)	16
Amazon EventBridge	4
AWS Identity and Access Management ruoli (IAM)	40
AWS Key Management Service (AWS KMS)	2
AWS Lambda	17
Amazon Simple Storage Service (Amazon S3)	2
Amazon Simple Notification Service (Amazon SNS)	2
Amazon Simple Queue Service (Amazon SQS)	2
AWS Systems Manager	62
AWS Step Functions	4

> **Important**
>
> Le risorse che vengono eliminate seguendo i passaggi indicati in questo schema non possono essere recuperate. Prima di seguire questi passaggi, verificate attentamente i nomi delle risorse e assicuratevi che siano stati creati da AFT.

Architecture

Il diagramma seguente mostra i componenti AFT e il flusso di lavoro di alto livello. AFT imposta una pipeline Terraform che ti aiuta a fornire e personalizzare i tuoi account. AWS Control Tower AFT segue un GitOps modello per automatizzare i processi di fornitura degli account. AWS Control Tower Si crea un file Terraform per una richiesta di account e lo si archivia in un repository, che fornisce l'input che attiva il flusso di lavoro AFT per il provisioning degli account. Una volta completata la fornitura dell'account, AFT può eseguire automaticamente ulteriori passaggi di personalizzazione.

In questa architettura:

- AWS Control Tower L'account di gestione Account AWS è un account dedicato al AWS Control Tower servizio. In genere viene anche chiamato conto di AWS pagamento o conto di AWS Organizations gestione.
- L'account AFT Management Account AWS è dedicato alle operazioni di gestione AFT. Si tratta di un account diverso dall'account di gestione dell'organizzazione.
- L'account Vended è un account Account AWS che contiene tutti i componenti e i controlli di base selezionati. AFT utilizza AWS Control Tower per vendere un nuovo account.

Per ulteriori informazioni su questa architettura, vedere Introduzione ad AFT nel AWS Control Tower workshop.

Tools (Strumenti)

Servizi AWS

- AWS Control Towerti aiuta a configurare e gestire un ambiente AWS multi-account, seguendo le migliori pratiche prescrittive.
- AWS Account Factory for Terraform (AFT) imposta una pipeline Terraform per aiutarti a fornire e personalizzare account e risorse. AWS Control Tower

- AWS Organizationsti aiuta a gestire e governare centralmente il tuo ambiente man mano che cresci e scalerai le tue risorse. AWS Utilizzando Organizations, puoi creare account e allocare risorse, raggruppare account per organizzare i flussi di lavoro, applicare politiche di governance e semplificare la fatturazione utilizzando un unico metodo di pagamento per tutti i tuoi account.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle. Questo modello richiede ruoli e autorizzazioni IAM.

Altri strumenti

- Terraform è uno strumento di infrastruttura come codice (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.

Best practice

- Per AWS Control Tower, consulta le migliori pratiche per AWS Control Tower gli amministratori nella documentazione. AWS Control Tower
- Per IAM, consulta le best practice di sicurezza nella documentazione IAM.

Epiche

Elimina le risorse AFT nell'account AFT Management

Operazione	Description	Competenze richieste
Eliminare le risorse identificate dal tag AFT.	1. Accedi all'account di gestione AFT con le autorizzazioni di amministratore. 2. Apri la AWS Resource Groups console. 3. Seleziona la regione in cui AWS Control Tower è stato distribuito.	Amministratore AWS, AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	4. Nel riquadro di navigazione, scegli Tag Editor. 5. Per i tipi di risorse, scegli Tutti i tipi di risorse supportati. 6. Per Tag, digitate managed_by come chiave del tag e AFT come valore del tag. 7. Scegli Cerca risorse. Questa ricerca mostra tutte le risorse create da AFT. 8. Identifica i nomi delle risorse ed eliminali utilizzando le console di servizio corrispondenti. Ad esempio, per eliminare le risorse di Parameter Store: a. Apri la AWS Systems Manager console. b. Nel riquadro di navigazione, selezionare Parameter Store (Archivio parametri). c. Nella casella di ricerca, fai clic per visualizzare il menu a discesa, scegli Nome, scegli equals, quindi digita /af. d. Eliminate i parametri in gruppi di 10. (Questo è il numero massimo che	

Operazione	Description	Competenze richieste
	puoi eliminare contemporaneamente). Per la versione AFT 1.12.2, ci saranno circa 62 risorse Parameter Store da eliminare. Tutti i nomi dei parametri inizieranno con /aft. Tuttavia, non tutte le risorse possono essere identificate da AWS Resource Groups. Nei passaggi seguenti, troverai ed eliminerai le risorse rimanenti.	
Elimina i ruoli IAM.	1. Accedi all'account di gestione AFT con le autorizzazioni di amministratore. 2. Apri la console IAM. 3. Elimina questi ruoli nell'ordine elencato (l'ordine è importante a causa delle dipendenze): • `aft-*` • `AWSAFTAdmin` • `AWSAFTExecution` • `AWSAFTService` • `codebuild_trigger_role`	Amministratore AWS, AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
Elimina il vault AWS Backup di backup.	1. Apri la AWS Backup console. 2. Individua l'archivio di backup denominato. `aws_backup_vault` 3. Verifica che il vault non contenga backup attivi. 4. Eliminare. `aws_backup_vault`	Amministratore AWS, AWS DevOps, DevOps ingegnere
Elimina CloudWatch le risorse Amazon.	1. Apri la CloudWatch console. 2. Elimina le seguenti risorse nell'ordine in cui sono elencate: a. Event bus: Elimina `aws_cloudwatch_event_bus` . b. Registri: cerca il prefisso AFT ed elimina tutti i gruppi di log correlati. c. Definizioni delle interrogazioni: Eliminare le seguenti interrogazioni: • `Customization Logs by Account ID` • `Customization Logs by Customization Request ID`	Amministratore AWS, AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
Eliminare AWS KMS risorse.	1. Passa alla regione secondaria, che funge da backend per il monitorag gio dello stato dello stato di AFT. 2. Apri la AWS KMS console. 3. Eliminare l'alias denominato AFT.	Amministratore AWS, AWS DevOps, DevOps ingegnere

Elimina le risorse AFT nell'account Log Archive

Operazione	Description	Competenze richieste
Eliminare i bucket S3.	1. Accedi all'account Log Archive con le autorizza zioni di amministratore. 2. Apri la console Amazon S3. 3. Svuota i seguenti secchi: • `aws-aft-logs-471112509802-us-east-1` • `aws-aft-s3-access-logs-471112509802-us-east-1` (`111122223333` Sostituis cilo con l'ID del tuo account). 4. Elimina i due bucket.	Amministratore AWS, AWS DevOps, DevOps ingegnere
Elimina i ruoli IAM.	1. Apri la console IAM.	Amministratore AWS, AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	2. Verifica che i seguenti ruoli non vengano utilizzati da alcun servizio attivo: • `AWSAFTService` • `AWSAFTExecution` 3. Eliminare i due ruoli.	

Elimina le risorse AFT nell'account Audit

Operazione	Description	Competenze richieste
Eliminare i ruoli IAM.	1. Accedi all'account Audit con le autorizzazioni di amministratore. 2. Apri la console IAM. 3. Verifica che i seguenti ruoli non vengano utilizzati da alcun servizio attivo: • `AWSAFTService` • `AWSAFTExecution` 4. Eliminare i due ruoli.	Amministratore AWS, AWS DevOps, DevOps ingegnere

Elimina le risorse AFT nell'account AWS Control Tower di gestione

Operazione	Description	Competenze richieste
Eliminare i ruoli IAM.	1. Accedi all'account di AWS Control Tower gestione con le autorizzazioni di amministratore. 2. Apri la console IAM.	Amministratore AWS, AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	3. Verifica che i seguenti ruoli non vengano utilizzati da alcun servizio attivo: • `AWSAFTService` • `AWSAFTExecution` • `aft-control-tower-events-rule` 4. Eliminare i tre ruoli.	
Eliminare EventBridge le regole.	1. Apri la EventBridge console Amazon. 2. Nel riquadro di navigazione a sinistra, selezionare Roles (Ruoli). 3. Trova e seleziona la regola denominata`aft-capture-ct-events` . 4. Scegli Elimina e conferma l'eliminazione quando richiesto.	Amministratore AWS, AWS DevOps, DevOps ingegnere

Risoluzione dei problemi

Problema	Soluzione
La disconnessione del gateway Internet non è riuscita.	Durante l'eliminazione di risorse identificate dal tag AFT, se riscontri questo problema quando scolleghi o elimini il gateway Internet, devi prima eliminare gli endpoint VPC: 1. Accedi all'account AFT Management, quindi apri la console Amazon VPC.

Problema	Soluzione
	2. Nel riquadro di navigazione, nell'elenco Filtra per VPC, scegli il VPC denominato. aft-management-vpc 3. Nel pannello di navigazione, seleziona Endpoint. 4. Seleziona gli endpoint associati al VPC. aft-management-vpc • Ricontrolla la colonna ID VPC prima dell'eliminazione per evitare di rimuovere gli endpoint sbagliati. • Fai attenzione a eliminare solo gli endpoint associati al VPC AFT. 5. Seleziona Actions (Operazioni), Delete VPC endpoints (Eliminazione di endpoint VPC). 6. Nella finestra di dialogo di conferma, digitate delete, quindi scegliete Elimina. 7. Attendi che lo stato dell'endpoint passi a Eliminato. Il completamento dell'eliminazione potrebbe richiedere alcuni minuti.

Problema	Soluzione
Non riesci a trovare le CloudWatch domande specificate.	Se non riesci a trovare le CloudWatch query create da AFT, procedi nel seguente modo: 1. Accedi all'account AFT Management, quindi apri la CloudWatch console. 2. Nel riquadro di navigazione, in Logs, scegli Logs Insights. 3. Nell'angolo in alto a destra, scegli l'icona Query salvate e di esempio. Ora dovresti essere in grado di vedere le interrogazioni AFT. Per uno screenshot, consultate la sezione Informazioni aggiuntive. 4. Seleziona le seguenti interrogazioni, quindi scegli Azioni, Elimina per rimuoverle. • `Customization Logs by Account ID` • `Customization Logs by Customization Request ID`

Risorse correlate

- GRASSO:
  - GitHub Repository
  - Officina
  - Documentazione
- AWS Control Tower documentazione

Informazioni aggiuntive

Per visualizzare le interrogazioni AFT nella dashboard di CloudWatch Logs Insights, scegli l'icona Query salvate e di esempio nell'angolo in alto a destra, come illustrato nella schermata seguente:

Crea una pipeline nelle regioni AWS che non supportano AWS CodePipeline

Anand Krishna Varanasi, Amazon Web Services

Riepilogo

Avviso: non AWS CodeCommit è più disponibile per i nuovi clienti. I clienti esistenti di AWS CodeCommit possono continuare a utilizzare il servizio normalmente. Ulteriori informazioni

AWS CodePipeline è un servizio di orchestrazione con distribuzione continua (CD) che fa parte di un set di DevOps strumenti di Amazon Web Services (AWS). Si integra con un'ampia varietà di fonti (come sistemi di controllo delle versioni e soluzioni di storage), prodotti e servizi di integrazione continua (CI) di AWS e partner AWS e prodotti open source per fornire un servizio di end-to-end flusso di lavoro per distribuzioni rapide di applicazioni e infrastrutture.

Tuttavia, CodePipeline non è supportato in tutte le regioni AWS ed è utile disporre di un orchestratore invisibile che collega i servizi AWS. CI/CD Questo modello descrive come implementare una pipeline end-to-end di flussi di lavoro nelle regioni AWS in cui CodePipeline non è ancora supportata utilizzando CI/CD servizi AWS come AWS CodeCommit CodeBuild, AWS e AWS CodeDeploy.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- CLI AWS Cloud Development Kit (AWS CDK) versione 2.28 o successiva

Architecture

Stack tecnologico Target

Il diagramma seguente mostra una pipeline creata in una regione che non supporta CodePipeline, ad esempio la regione africana (Città del Capo). Uno sviluppatore invia i file di CodeDeploy configurazione (chiamati anche script hook di deployment lifecycle) al repository Git ospitato da. CodeCommit (Vedi l'GitHub archivio fornito con questo modello). Viene avviata automaticamente una EventBridge regola Amazon. CodeBuild

I file di CodeDeploy configurazione vengono recuperati CodeCommit come parte della fase di origine della pipeline e trasferiti in. CodeBuild

Nella fase successiva, CodeBuild esegue le seguenti attività:

1. Scarica il file TAR del codice sorgente dell'applicazione. Puoi configurare il nome di questo file utilizzando Parameter Store, una funzionalità di AWS Systems Manager.
2. Scarica i file CodeDeploy di configurazione.
3. Crea un archivio combinato di codice sorgente dell'applicazione e file di CodeDeploy configurazione specifici per il tipo di applicazione.
4. Avvia la CodeDeploy distribuzione su un'istanza Amazon Elastic Compute Cloud (Amazon EC2) utilizzando l'archivio combinato.

Tools (Strumenti)

Servizi AWS

- AWS CodeBuild è un servizio di build completamente gestito che ti aiuta a compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS CodeCommit è un servizio di controllo delle versioni che consente di archiviare e gestire archivi Git in modo privato, senza dover gestire il proprio sistema di controllo del codice sorgente.
- AWS CodeDeploy automatizza le distribuzioni su Amazon EC2 o istanze locali, funzioni AWS Lambda o servizi Amazon Elastic Container Service (Amazon ECS).
- AWS ti CodePipeline aiuta a modellare e configurare rapidamente le diverse fasi di un rilascio di software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software che aiuta a definire e fornire l'infrastruttura cloud AWS in codice.

Codice

Il codice per questo modello è disponibile nel repository GitHub CodePipeline Unsupported Regions.

Epiche

Configura la tua postazione di lavoro per sviluppatori

Operazione	Description	Competenze richieste
Installa l'interfaccia a riga di comando di AWS CDK.	Per istruzioni, consulta la documentazione di AWS CDK.	AWS DevOps
Installa un client Git.	Per creare i commit, puoi usare un client Git installato sul tuo computer locale e poi inviare i commit al CodeCommit repository. Per eseguire la configurazione CodeCommit con il tuo client Git, consulta la CodeCommit documentazione.	AWS DevOps
Installa npm.	Installa il gestore di pacchetti npm. Per ulteriori informazioni, consulta la documentazione di npm.	AWS DevOps

Configura la pipeline

Operazione	Description	Competenze richieste
Clona il repository del codice.	Clona l'archivio delle regioni GitHub CodePipeline non supportate sul computer locale eseguendo il comando seguente. `git clone https://github.com/aws-samples/invisible-code`	DevOps ingegnere

Operazione	Description	Competenze richieste
	`pipeline-unsupported-regions`	

Operazione	Description	Competenze richieste
Imposta i parametri in cdk.json.	Apri il `cdk.json` file e fornisci i valori per i seguenti parametri : `"pipeline_account" :"XXXXXXXXXXXX", "pipeline_region":" us-west-2", "repo_name": "app-dev- repo", "ec2_tag_key": "test- vm", "configName" : "cbdeployconfig", "deploymentGroupNa me": "cbdeploygroup", "applicationName" : "cbdeployapplicati on", "projectName" : "CodeBuildProject"` dove: • `pipeline_account` è l'account AWS in cui verrà creata la pipeline. • `pipeline_region` è la regione AWS in cui verrà costruita la pipeline. • `repo_name` è il nome del CodeCommit repository. • `ec2_tag_key` è il tag allegato all'istanza EC2 su cui vuoi distribuire il codice.	AWS DevOps

Operazione	Description	Competenze richieste
	• `configName` è il nome del CodeDeploy file di configurazione. • `deploymentGroupName` è il nome del gruppo CodeDeploy di distribuzione. • `applicationName` è il nome CodeDeploy dell'applicazione. • `projectName` è il nome CodeBuild del progetto.	

Configura la libreria di costruzioni AWS CDK.	Nel GitHub repository clonato, usa i seguenti comandi per installare la libreria AWS CDK construct, creare l'applicazione e sintetizzarla per generare il modello AWS per l'applicazione. CloudFormation
	`npm i aws-cdk-lib`
	`npm run build`
	`cdk synth`
		AWS DevOps

Distribuisci l'applicazione AWS CDK di esempio.	Distribuisci il codice eseguendo il seguente comando in una regione non supportata (ad esempio). `af-south-1` `cdk deploy`	AWS DevOps

Configura il CodeCommit repository per CodeDeploy

Operazione	Description	Competenze richieste
Configurazione CI/CD per l'applicazione.	Clona il CodeCommit repository specificato nel `cdk.json` file (questo è chiamato di `app-dev-repo` default) per configurare la CI/CD pipeline per l'applicazione. `git clone https://git-codecommit.us-west-2.amazonaws.com/v1/repos/app-dev-repo` dove il nome e la regione del repository dipendono dai valori forniti nel file. `cdk.json`	AWS DevOps

Testa la pipeline

Operazione	Description	Competenze richieste
Testa la pipeline con le istruzioni di implementazione.	La `CodeDeploy_Files` cartella del repository GitHub CodePipeline Unsupported Regions include file di esempio che indicano come distribuire l'applicazione CodeDeploy . Il `appspec.yml` file è un file di CodeDeploy configurazione che contiene gli hook per controllare il flusso di distribuzione delle applicazioni. È possibile utilizzare i	AWS DevOps

Operazione	Description	Competenze richieste
	file di esempio`index.html` , `start_server.sh` `stop_server.sh` , e `install_dependencies.sh` aggiornare un sito Web ospitato su Apache. Questi sono esempi: puoi utilizzare il codice nel GitHub repository per distribuire qualsiasi tipo di applicazione. Quando i file vengono inviati al CodeCommit repository, la pipeline invisibile viene avviata automaticamente. Per i risultati di implementazione, controlla i risultati delle singole fasi nelle console e. CodeBuild CodeDeploy	

Risorse correlate

- Guida introduttiva (documentazione AWS CDK)
- Introduzione al Cloud Development Kit (CDK) (AWS Workshop Studio)
- Workshop CDK AWS

Personalizza i nomi dei ruoli predefiniti utilizzando AWS CDK gli aspetti e gli escape hatch

SANDEEP SINGH e James Jacob, Amazon Web Services

Riepilogo

Questo modello dimostra come personalizzare i nomi predefiniti dei ruoli creati dai costrutti. AWS Cloud Development Kit (AWS CDK) La personalizzazione dei nomi dei ruoli è spesso necessaria se l'organizzazione ha vincoli specifici basati sulle convenzioni di denominazione. Ad esempio, l'organizzazione potrebbe impostare limiti di autorizzazioni AWS Identity and Access Management (IAM) o politiche di controllo dei servizi (SCPs) che richiedono un prefisso specifico nei nomi dei ruoli. In questi casi, i nomi di ruolo predefiniti generati dai AWS CDK costrutti potrebbero non soddisfare queste convenzioni e potrebbero dover essere modificati. Questo modello soddisfa tali requisiti utilizzando le vie di fuga e gli aspetti di. AWS CDK Gli escape hatch vengono utilizzati per definire nomi di ruolo personalizzati e aspetti per applicare un nome personalizzato a tutti i ruoli, in modo da garantire il rispetto delle politiche e dei vincoli dell'organizzazione.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Prerequisiti specificati nella documentazione AWS CDK

Limitazioni

- Gli aspetti filtrano le risorse in base ai tipi di risorse, quindi tutti i ruoli condividono lo stesso prefisso. Se sono necessari prefissi di ruolo diversi per ruoli diversi, è necessario filtrare ulteriormente in base ad altre proprietà. Ad esempio, per assegnare prefissi diversi ai ruoli associati alle AWS Lambda funzioni, è possibile filtrare per attributi o tag di ruolo specifici e applicare un prefisso per i ruoli relativi a Lambda e un prefisso diverso per altri ruoli.
- I nomi di ruolo IAM hanno una lunghezza massima di 64 caratteri, pertanto i nomi di ruolo modificati devono essere tagliati per soddisfare questa restrizione.
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Architecture

Stack tecnologico Target

- AWS CDK
- AWS CloudFormation

Architettura di destinazione

- Un' AWS CDK app è composta da uno o più CloudFormation stack, che vengono sintetizzati e distribuiti per gestire le risorse. AWS
- Per modificare una proprietà di una risorsa AWS CDK gestita che non è esposta da un costrutto di livello 2 (L2), utilizzate un escape hatch per sovrascrivere le CloudFormation proprietà sottostanti (in questo caso, il nome del ruolo) e un aspetto per applicare il ruolo a tutte le risorse dell'app durante il processo di sintesi dello stack. AWS CDK AWS CDK

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che consente di definire e fornire l'infrastruttura nel codice. Cloud AWS
- AWS CDK Command Line Interface (AWS CDK CLI) (nota anche come AWS CDK Toolkit) è un kit di sviluppo cloud a riga di comando che ti aiuta a interagire con la tua app. AWS CDK Il `cdk` comando CLI è lo strumento principale per interagire con l'app. AWS CDK Esegue l'app, interroga il modello di applicazione definito e produce e distribuisce i CloudFormation modelli generati da. AWS CDK
- CloudFormationti aiuta a configurare AWS le risorse, a fornirle in modo rapido e coerente e a gestirle durante il loro ciclo di vita in tutte le regioni. Account AWS

Archivio di codice

Il codice sorgente e i modelli per questo pattern sono disponibili nel repository GitHub CDK Aspects Override.

Best practice

Consultate le migliori pratiche per l'utilizzo di AWS CDK in per TypeScript creare progetti IaC sul sito web di Prescriptive Guidance. AWS

Epiche

Installa la AWS CDK CLI

Operazione	Description	Competenze richieste
Installa la AWS CDK CLI.	Per installare la AWS CDK CLI a livello globale, esegui il comando: `npm install -g aws-cdk`	AWS DevOps
Verifica la versione.	Eseguire il comando : `cdk --version` Conferma di utilizzare la versione 2 della AWS CDK CLI.	AWS DevOps
Avvia l' AWS CDK ambiente.	Prima di distribuire i CloudFormation modelli, preparate l'account e Regione AWS quello che desiderat e utilizzare. Eseguire il comando : `cdk bootstrap <account> /<Region>` Per ulteriori informazi oni, consulta il AWS CDK	AWS DevOps

Operazione	Description	Competenze richieste
	bootstrap nella documentazione. AWS	

Implementa l' AWS CDK app per dimostrare l'uso degli aspetti

Operazione	Description	Competenze richieste

Configura il progetto.	1. Clona il GitHub repository per questo pattern sul tuo computer locale:
	`git clone https://github.com/aws-samples/cdk-aspects-override`
	2. Passa alla directory del progetto sul tuo computer locale.
	3. Installa le dipendenze del progetto:
	`npm ci`
		AWS DevOps

Distribuisci stack con nomi di ruolo predefiniti assegnati da. AWS CDK	Implementa due CloudFormation stack (`ExampleStack1` and`ExampleStack2`) che contengono le funzioni Lambda e i ruoli associati: `npm run deploy:ExampleAppWithoutAspects`	AWS DevOps

Operazione	Description	Competenze richieste
	Il codice non passa esplicita mente le proprietà dei ruoli, quindi i nomi dei ruoli verranno costruiti da. AWS CDK Per un esempio di output, consulta la sezione Informazi oni aggiuntive.	

Operazione	Description	Competenze richieste

Implementa stack con aspetti.	In questo passaggio, applichi un aspetto che impone una convenzione sui nomi dei ruoli aggiungendo un prefisso a tutti i ruoli IAM distribuiti nel progetto. AWS CDK L'aspetto è definito nel file. `lib/aspects.ts` L'aspetto utilizza una via di fuga per sovrascrivere il nome del ruolo aggiungendo un prefisso. L'aspetto viene applicato agli stack del file. `bin/app-with-aspects.ts` Il prefisso del nome del ruolo utilizzato in questo esempio è. `dev-unicorn`
	
	1. Modificare il file `bin/app-with-aspects.ts` .
	2. Nel file, aggiorna la variabile `ROLE_NAME_PREFIX` con il prefisso`dev-unicorn` :
	
	`const app = new cdk.App();`
	
	`// Define a prefix for the role names`
	`const ROLE_NAME_PREFIX = 'dev-unicorn';`
	
	`// Instantiate the RoleNamingConventionAspect with the desired prefix`
		AWS DevOps

Operazione	Description	Competenze richieste

	`const roleNamingConventionAspect = new RoleNamingConventionAspect(ROLE_NAME_PREFIX);`
	3. Implementa l' AWS CDK app con alcuni aspetti:
	`npm run deploy:ExampleAppWithAspects`
	Per un esempio di output, consulta la sezione Informazioni aggiuntive.	

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Elimina i tuoi AWS CloudFormation stack.	Dopo aver finito di utilizzare questo pattern, esegui il comando seguente per ripulire le risorse ed evitare costi aggiuntivi: `cdk destroy --all -f && cdk --app npx ts-node bin/app-with-aspects.ts' destroy --all -f`	AWS DevOps

risoluzione dei problemi

Problema	Soluzione
Si verificano problemi nell'utilizzo di AWS CDK.	Vedi Risoluzione dei AWS CDK problemi più comuni nella AWS CDK documentazione.

Risorse correlate

- AWS Cloud Development Kit (AWS CDK)
- AWS CDK documentazione
- AWS CDK su GitHub
- boccaporti di fuga
- Aspetti e AWS CDK

Informazioni aggiuntive

Nomi di ruolo creati da CloudFormation without aspects

```
Outputs:
ExampleStack1WithoutAspects.Function1RoleName = example-stack1-without-as-
Function1LambdaFunctionSe-y7FYTY6FXJXA
ExampleStack1WithoutAspects.Function2RoleName = example-stack1-without-as-
Function2LambdaFunctionSe-dDZV4rkWqWnI
...

Outputs:
ExampleStack2WithoutAspects.Function3RoleName = example-stack2-without-as-
Function3LambdaFunctionSe-ygMv49iTyMq0
```

Nomi di ruolo creati da CloudFormation con aspetti

```
Outputs:
ExampleStack1WithAspects.Function1RoleName = dev-unicorn-
Function1LambdaFunctionServiceRole783660DC
ExampleStack1WithAspects.Function2RoleName = dev-unicorn-
Function2LambdaFunctionServiceRole2C391181
...
```

```
Outputs:
ExampleStack2WithAspects.Function3RoleName = dev-unicorn-
Function3LambdaFunctionServiceRole4CAA721C
```

Implementa un cluster Cassandra su Amazon EC2 con dati statici privati IPs per evitare il ribilanciamento

Dipin Jain, Amazon Web Services

Riepilogo

L'IP privato di un'istanza Amazon Elastic Compute Cloud (Amazon EC2) viene mantenuto per tutto il suo ciclo di vita. Tuttavia, l'IP privato potrebbe cambiare durante un arresto anomalo del sistema pianificato o non pianificato, ad esempio durante un aggiornamento di Amazon Machine Image (AMI). In alcuni scenari, il mantenimento di un IP statico privato può migliorare le prestazioni e i tempi di ripristino dei carichi di lavoro. Ad esempio, l'utilizzo di un IP statico per un nodo iniziale di Apache Cassandra impedisce al cluster di incorrere in un sovraccarico di ribilanciamento.

Questo modello descrive come collegare un'interfaccia elastica di rete secondaria alle EC2 istanze per mantenere l'IP statico durante il rehosting. Il modello si concentra sui cluster Cassandra, ma è possibile utilizzare questa implementazione per qualsiasi architettura che tragga vantaggio dalla staticità privata. IPs

Prerequisiti e limitazioni

Prerequisiti

- Un account Amazon Web Service (AWS) attivo

Versioni del prodotto

- DataStax versione 5.11.1
- Sistema operativo: Ubuntu 16.04.6 LTS

Architecture

Architettura di origine

La fonte potrebbe essere un cluster Cassandra su una macchina virtuale (VM) locale o su EC2 istanze nel cloud AWS. Il diagramma seguente illustra il secondo scenario. Questo esempio include quattro nodi del cluster: tre nodi iniziali e un nodo di gestione. Nell'architettura di origine, ogni nodo ha una singola interfaccia di rete collegata.

Architettura Target

Il cluster di destinazione è ospitato su EC2 istanze con un'interfaccia elastica di rete secondaria collegata a ciascun nodo, come illustrato nel diagramma seguente.

Automazione e scalabilità

Puoi anche automatizzare il collegamento di una seconda interfaccia di rete elastica a un gruppo di Auto EC2 Scaling, come descritto in un video dell'AWS Knowledge Center.

Epiche

Configura un cluster Cassandra su Amazon EC2

Operazione	Description	Competenze richieste
Avvia EC2 nodi per ospitare un cluster Cassandra.	Sulla EC2 console Amazon, avvia quattro EC2 istanze per i tuoi nodi Ubuntu nel tuo account AWS. Tre nodi (seed) vengono utilizzati per il cluster Cassandra e il quarto nodo funge da nodo di gestione del cluster in cui installerai DataStax Enterprise (DSE). OpsCenter Per istruzioni, consulta la EC2 documentazione di Amazon.	Tecnico del cloud
Conferma le comunicazioni tra i nodi.	Assicurati che i quattro nodi possano comunicare tra loro tramite le porte di gestione del database e del cluster.	Ingegnere di rete
Installa DSE OpsCenter sul nodo di gestione.	Installa DSE OpsCenter 6.1 dal pacchetto Debian sul nodo di gestione. Per le istruzioni,	DBA

Operazione	Description	Competenze richieste
	consultate la documentazione. DataStax	

Operazione	Description	Competenze richieste
Crea un'interfaccia di rete secondaria.	Cassandra genera un identificatore univoco universale (UUID) per ogni nodo in base all'indirizzo IP dell' EC2 istanza di quel nodo. Questo UUID viene utilizzato per distribuire nodi virtuali (vnodes) sull'anello. Quando Cassandra viene distribuito su EC2 istanze, gli indirizzi IP vengono assegnati automaticamente alle istanze man mano che vengono create. In caso di interruzione pianificata o non pianificata, l'indirizzo IP della nuova EC2 istanza cambia, la distribuzione dei dati cambia e l'intero anello deve essere ribilanciato. Questo non è auspicabile. Per conservare l'indirizzo IP assegnato, utilizza un'interfaccia elastica di rete secondaria con un indirizzo IP fisso. 1. Sulla EC2 console Amazon, scegli Interfacce di rete, Crea interfaccia di rete. 2. Per Subnet, seleziona la sottorete in cui hai creato l' EC2 istanza.	Tecnico del cloud

Operazione	Description	Competenze richieste
	3. Per IPv4 Indirizzo privato, scegli Assegnazione automatica. 4. Per Gruppi di sicurezza , seleziona un gruppo di sicurezza, quindi scegli Crea interfaccia di rete. Per ulteriori informazioni sulla creazione di un'interf accia di rete, consulta la EC2 documentazione di Amazon.	
Collega l'interfaccia di rete secondaria ai nodi del cluster.	1. Sulla EC2 console Amazon, scegli Istanze. 2. Seleziona la casella di controllo relativa all' EC2 istanza che hai creato in precedenza. 3. Scegliere Actions (Operazio ni), Networking (Reti), Attach network interface (Collega interfaccia di rete). 4. Seleziona l'interfaccia di rete creata nel passaggio precedente, quindi scegli Allega. Per ulteriori informazioni sul collegamento di un'interf accia di rete, consulta la EC2 documentazione di Amazon.	Ingegnere del cloud

Operazione	Description	Competenze richieste
Aggiungi percorsi in Amazon EC2 per risolvere il problema del routing asimmetrico.	Quando colleghi la seconda interfaccia di rete, è molto probabile che la rete esegua un routing asimmetrico. Per evitare ciò, è possibile aggiungere percorsi per le nuove interfacce di rete. Per una spiegazione approfondita e la correzione del routing asimmetrico, guarda il video dell'AWS Knowledge Center o Overcoming Asymmetric Routing on Multi-Home Servers (articolo nel Linux Journal di Patrick, 5 aprile 2004). McManus	Ingegnere di rete
Aggiorna le voci DNS in modo che puntino all'IP dell'interfaccia di rete secondaria.	Indirizza il nome di dominio completo (FQDN) del nodo all'IP dell'interfaccia di rete secondaria.	Ingegnere di rete
Installa e configura il cluster Cassandra utilizzando OpsCenter DSE.	Quando i nodi del cluster sono pronti con le interfacce di rete secondarie, puoi installare e configurare il cluster Cassandra.	DBA

Ripristina il cluster da un guasto del nodo

Operazione	Description	Competenze richieste
Crea un AMI per il nodo seed del cluster.	Effettua un backup dei nodi in modo da poterli ripristinare con i file binari del database in caso di errore del nodo. Per istruzioni, consulta Creare un'AMI nella EC2 documenta zione di Amazon.	Amministratore di backup
Ripristina in caso di guasto del nodo.	Sostituisci il nodo guasto con una nuova EC2 istanza lanciata dall'AMI e collega l'interfaccia di rete secondaria del nodo guasto.	Amministratore di backup
Verifica che il cluster Cassandra sia integro.	Quando il nodo sostitutivo è attivo, verifica lo stato del cluster in OpsCenter DSE.	DBA

Risorse correlate

- Installazione di DSE OpsCenter 6.1 dal pacchetto Debian (documentazione) DataStax
- Come far funzionare un'interfaccia di rete secondaria in un' EC2 istanza di Ubuntu (video AWS Knowledge Center)
- Best practice per l'esecuzione di Apache Cassandra su Amazon (post sul blog EC2 AWS)

Estendi VRFs ad AWS utilizzando AWS Transit Gateway Connect

Adam Till, Yashar Araghi, Vikas Dewangan e Mohideen, Amazon Web Services HajaMohideen

Riepilogo

Il routing e l'inoltro virtuali (VRF) sono una funzionalità delle reti tradizionali. Utilizza domini di routing logico isolati, sotto forma di tabelle di routing, per separare il traffico di rete all'interno della stessa infrastruttura fisica. Puoi configurare AWS Transit Gateway per supportare l'isolamento VRF quando connetti la tua rete locale ad AWS. Questo modello utilizza un'architettura di esempio per connettere gli ambienti locali VRFs a diverse tabelle di routing dei gateway di transito.

Questo modello utilizza interfacce virtuali di transito (VIFs) in AWS Direct Connect e gli allegati Transit Gateway Connect per estendere il. VRFs Un VIF di transito viene utilizzato per accedere a uno o più gateway di transito Amazon VPC associati ai gateway Direct Connect. Un allegato Transit Gateway Connect collega un gateway di transito con un'appliance virtuale di terze parti in esecuzione in un VPC. Un allegato Transit Gateway Connect supporta il protocollo di tunnel Generic Routing Encapsulation (GRE) per prestazioni elevate e supporta il Border Gateway Protocol (BGP) per il routing dinamico.

L'approccio descritto in questo modello presenta i seguenti vantaggi:

- Utilizzando Transit Gateway Connect, puoi pubblicizzare fino a 1.000 rotte sul peer Transit Gateway Connect e ricevere fino a 5.000 rotte da esso. L'utilizzo della funzionalità VIF di transito Direct Connect senza Transit Gateway Connect è limitato a 20 prefissi per gateway di transito.
- Puoi mantenere l'isolamento del traffico e utilizzare Transit Gateway Connect per fornire servizi ospitati su AWS, indipendentemente dagli schemi di indirizzi IP utilizzati dai tuoi clienti.
- Il traffico VRF non deve necessariamente attraversare un'interfaccia virtuale pubblica. Ciò semplifica il rispetto dei requisiti di conformità e sicurezza in molte organizzazioni.
- Ogni tunnel GRE supporta fino a 5 Gbps e puoi avere fino a quattro tunnel GRE per ogni collegamento Connect del gateway di transito. È più veloce di molti altri tipi di connessione, come le connessioni AWS Site-to-Site VPN che supportano fino a 1,25 Gbps.

Prerequisiti e limitazioni

Prerequisiti

- Gli account AWS richiesti sono stati creati (consulta l'architettura per i dettagli)

- Autorizzazioni per assumere un ruolo AWS Identity and Access Management (IAM) in ogni account.
- I ruoli IAM in ogni account devono disporre delle autorizzazioni per effettuare il provisioning delle risorse AWS Transit Gateway e AWS Direct Connect. Per ulteriori informazioni, consulta Autenticazione e controllo degli accessi per i gateway di transito e Vedi Gestione delle identità e degli accessi per Direct Connect.
- Le connessioni Direct Connect sono state create correttamente. Per ulteriori informazioni, vedere Creare una connessione utilizzando la procedura guidata di connessione.

Limitazioni

- Esistono dei limiti per gli allegati del gateway di transito VPCs agli account di produzione, controllo qualità e sviluppo. Per ulteriori informazioni, consulta Transit gateway attachments to a VPC.
- Esistono dei limiti per la creazione e l'utilizzo di gateway Direct Connect. Per ulteriori informazioni, consulta le quote di AWS Direct Connect.

Architecture

Architettura Target

La seguente architettura di esempio fornisce una soluzione riutilizzabile per implementare il transito VIFs con gli allegati Transit Gateway Connect. Questa architettura offre resilienza utilizzando più postazioni Direct Connect. Per ulteriori informazioni, consulta Resilienza massima nella documentazione di Direct Connect. La rete locale prevede produzione, controllo qualità e sviluppo VRFs estesi ad AWS e isolati utilizzando tabelle di routing dedicate.

Nell'ambiente AWS, due account sono dedicati all'estensione di VRFs: un account Direct Connect e un account hub di rete. L'account Direct Connect contiene la connessione e il transito VIFs per ogni router. I transiti vengono creati VIFs dall'account Direct Connect ma li si distribuisce all'account dell'hub di rete in modo da poterli associare al gateway Direct Connect nell'account dell'hub di rete. L'account dell'hub di rete contiene il gateway Direct Connect e il gateway di transito. Le risorse AWS sono collegate come segue:

1. Transit VIFs collega i router nelle sedi Direct Connect con AWS Direct Connect nell'account Direct Connect.
2. Un transito VIF collega Direct Connect al gateway Direct Connect nell'account dell'hub di rete.

3. Un'associazione di gateway di transito collega il gateway Direct Connect con il gateway di transito nell'account dell'hub di rete.
4. Gli allegati Transit gateway Connect collegano il gateway di transito con gli account di produzione, controllo qualità e sviluppo. VPCs

Architettura Transit VIF

Il diagramma seguente mostra i dettagli di configurazione per il transito. VIFs Questa architettura di esempio utilizza una VLAN per la sorgente del tunnel, ma è possibile utilizzare anche un loopback.

Di seguito sono riportati i dettagli di configurazione, come i numeri di sistema autonomi (ASNs), per il transito. VIFs

Risorsa	Elemento	Dettaglio
router-01	ASN	65534
router-02	ASN	65534
router-03	ASN	65534
router-04	ASN	65534
Gateway Direct Connect	ASN	64601
Transit Gateway	ASN	64600
	blocco CIDR	10,100254,0/24

Architettura Transit gateway Connect

Il diagramma e le tabelle seguenti descrivono come configurare un singolo VRF tramite un allegato Transit Gateway Connect. Inoltre VRFs, assegna tunnel, gateway di transito IDs, indirizzi IP GRE e BGP univoci all'interno dei blocchi CIDR. L'indirizzo IP GRE peer corrisponde all'indirizzo IP peer del router dal VIF di transito.

La tabella seguente contiene i dettagli di configurazione del router.

Router	Tunnel	IP address (Indirizzo IP)	Origine	Destinazione
router-01	Tunnel 1	169,254,101,17	VLAN 60 169,254,1001	10,100,254,1
router-02	Tunnel 11	169,254,101,81	VLAN 61 169,254,100,5	10,100254,11
router-03	Tunnel 21	169,254,101,145	VLAN 62 169,254,100,9	10,100254,21
router-04	Tunnel 31	169,254,101,209	VLAN 63 169,254,100,13	10,100254,31

La tabella seguente contiene i dettagli di configurazione del gateway di transito.

Tunnel	Indirizzo IP GRE del gateway di transito	Indirizzo IP GRE peer	BGP all'interno dei blocchi CIDR
Tunnel 1	10.100.254,1	VLAN 60 169,254,1001	169,254,101,16/29
Tunnel 11	10.100.254,11	VLAN 61 169,254,100,5	169,254,101,80/29
Tunnel 21	10.100.254,21	VLAN 62 169,254,100,9	169,254,101,144/29
Tunnel 31	10.100.254,31	VLAN 63 169,254,100,13	169,254,101,208/29

Distribuzione

La sezione Epics descrive come implementare una configurazione di esempio per un singolo VRF su più router clienti. Una volta completati i passaggi da 1 a 5, puoi creare nuovi allegati Transit Gateway Connect utilizzando i passaggi 6—7 per ogni nuovo VRF che stai estendendo in AWS:

1. Crea il gateway di transito.
2. Crea una tabella di routing Transit Gateway per ogni VRF.
3. Crea le interfacce virtuali di transito.
4. Crea il gateway Direct Connect.
5. Crea l'interfaccia virtuale del gateway Direct Connect e le associazioni dei gateway con prefissi consentiti.
6. Crea l'allegato Transit Gateway Connect.
7. Crea i peer Transit Gateway Connect.
8. Associate l'allegato Transit Gateway Connect alla tabella delle rotte.
9. Pubblicizza i percorsi verso i router.

Tools (Strumenti)

Servizi AWS

- AWS Direct Connect collega la rete interna a una posizione Direct Connect tramite un cavo Ethernet standard in fibra ottica. Con questa connessione, puoi creare interfacce virtuali direttamente ai servizi AWS pubblici bypassando i provider di servizi Internet nel tuo percorso di rete.
- AWS Transit Gateway è un hub centrale che collega cloud privati virtuali (VPCs) e reti locali.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito. Questa rete virtuale è simile a una rete tradizionale che gestiresti nel tuo data center, con i vantaggi dell'utilizzo dell'infrastruttura scalabile di AWS.

Epiche

Pianifica l'architettura

Operazione	Description	Competenze richieste
Crea diagrammi di architettura personalizzati.	1. Nella sezione Allegati, scarica il modello di diagramma. 2. Aprire il diagramma allegato in Microsoft Office PowerPoint. 3. Nella diapositiva Panoramica dell'architettura, personalizza il diagramma dell'architettura per il tuo ambiente. Identifica gli ambienti locali VRFs che devono essere estesi nel tuo ambiente AWS. 4. Nella diapositiva Transit VIF, personalizza il diagramma dell'architettura. Identifica i numeri AS dei router, del gateway Direct Connect e del gateway di transito. Identifica gli indirizzi IP a ciascuna estremità del VIF di transito. 5. Nella diapositiva Transit Gateway Connect, personalizzate un diagramma di architettura per ogni VRF. Identifica tutti gli indirizzi IP richiesti necessari per configura	Architetto cloud, amministratore di rete

Operazione	Description	Competenze richieste
	re i router e i peer Transit Gateway Connect.	

Creare le risorse Transit Gateway

Operazione	Description	Competenze richieste
Crea il gateway di transito.	1. Accedi all'account dell'hub di rete. 2. Segui le istruzioni riportate in Creare un gateway di transito. Nota quanto segue per questo modello: • Per il numero di sistema autonomo (ASN) lato Amazon, inserisci un ASN univoco. Ai fini di questo esempio, l'ASN è. `64600` • Seleziona il supporto DNS. • Per questa architettura di esempio, non sono richiesti il supporto VPN ECMP, l'associazione della tabella di routing predefinita, la proroga della tabella di routing predefinita e il supporto Multicast. • Per i blocchi CIDR del gateway di transito, inserisci i blocchi CIDR	Amministratore di rete, architetto del cloud

Operazione	Description	Competenze richieste
	per il tuo gateway di IPv4 transito. Ai fini di questo esempio, il blocco CIDR è. `10.100.254.0/24`	
Crea la tabella delle rotte del gateway di transito.	Segui le istruzioni riportate in Creare una tabella di routing del gateway di transito. Per questo modello, tenete presente quanto segue: • Per il tag Name, fornite un nome per la tabella delle rotte del gateway di transito. Ti consigliamo di utilizzare un nome che corrisponda al VRF, ad esempio`routetable-dev-vrf` . • Per Transit gateway ID, scegli il gateway di transito che hai creato in precedenza.	Architetto del cloud, amministratore di rete

Crea le interfacce virtuali di transito

Operazione	Description	Competenze richieste
Crea le interfacce virtuali di transito.	1. Accedi all'account Direct Connect. 2. Segui le istruzioni riportate in Creare un'interfaccia virtuale di transito per il gateway Direct Connect.	Architetto del cloud, amministratore di rete

Operazione	Description	Competenze richieste
	Per questo modello, tenete presente quanto segue: • Per il nome dell'interfaccia virtuale, immettete un nome per il file VIF di transito. Si consiglia di utilizzare un nome che corrisponda al router, ad esempio `transit-vif-router01` . • Per Connessione, seleziona il router, ad esempio `router-01` . • Per Proprietario dell'interfaccia virtuale, immettere l'ID dell'account dell'hub di rete. Per istruzioni, consulta Visualizza l'ID del tuo account AWS. • Per il gateway Direct Connect, non effettuare alcuna selezione. Il gateway Direct Connect viene collegato in un passaggio successivo. • Per VLAN, inserisci la VLAN del router, ad esempio. `60` • Per BGP ASN, inserisci l'ASN del router, ad esempio. `65534` • In Impostazioni aggiuntive, procedi come segue:	

Operazione	Description	Competenze richieste
	• Scegli IPv4. • Per il peer ip del router, inserisci l'indirizzo IP peer del router, ad esempio. `169.254.100.1` • Per Amazon router peer ip, inserisci l'IP peer del router Amazon, ad esempio. `169.254.100.2` • Per la chiave di autenticazione BGP, è richiesta una password. Se questo campo viene lasciato vuoto, AWS crea una chiave accessibile solo in questo account. 3. Ripeti queste istruzioni per creare tutti i transiti VIFs per il VRF.	

Crea le risorse Direct Connect

Operazione	Description	Competenze richieste
Creare un gateway Direct Connect.	1. Accedere all'account dell'hub di rete. 2. Segui le istruzioni riportate in Creazione di un gateway Direct Connect. Per questo	Architetto del cloud, amministratore di rete

Operazione	Description	Competenze richieste
	modello, tenete presente quanto segue: • Per l'ASN lato Amazon, inserisci l'ASN del gateway Direct Connect, ad esempio. `64601` • Non scegliere un gateway privato virtuale.	
Collega il gateway Direct Connect al transito VIFs.	1. Nell'account dell'hub di rete, apri la console AWS Direct Connect alla versione https://console.aws.amazon.com/directconnect/v2/. 2. Nel riquadro di navigazione, scegli Virtual Interfaces (Interfacce virtuali). 3. Seleziona un nuovo file VIF di transito, quindi scegli Accept. 4. Scegli il gateway Direct Connect che hai creato. 5. Ripeti queste istruzioni per ogni VIF di transito.	Architetto del cloud, amministratore di rete

Operazione	Description	Competenze richieste
Crea le associazioni del gateway Direct Connect con i prefissi consentiti.	Nell'account dell'hub di rete, segui le istruzioni in Per associare un gateway di transito. Per questo modello, tenete presente quanto segue: • Per i gateway, scegli il gateway di transito che hai creato in precedenza. • Per i prefissi consentiti, inserisci il blocco CIDR assegnato al gateway di transito, ad esempio. **`10.100.254.0/24`** La creazione di questa associazione crea automatic amente un allegato Transit Gateway con un tipo di risorsa Direct Connect Gateway. Non è necessario che questo allegato sia associato a una tabella di routing del gateway di transito.	Architetto del cloud, amministr atore di rete

Operazione	Description	Competenze richieste
Crea l'allegato Transit Gateway Connect.	1. Nell'account dell'hub di rete, apri la console Amazon VPC all'indirizzo. https://console.aws.amazon.com/vpc/ 2. Nel riquadro di navigazione, seleziona Collegamenti del gateway di transito alla VPN. 3. Selezionare Create transit gateway attachments (crea collegamenti del gateway di transito). 4. Per il tag Name, inserisci un nome per l'allegato. Ti consigliamo di utilizzare un nome che corrisponda al VRF, ad esempio`PROD-VRF`. 5. Per Transit gateway ID, scegli il gateway di transito che hai creato in precedenza. 6. In Tipo collegamento, seleziona Connect. 7. Per Transport attachment ID, scegliete il gateway Direct Connect creato in precedenza. 8. Selezionare Create Transit Gateway Attachments (Crea collegamenti del gateway di transito).	Architetto del cloud, amministratore di rete

Operazione	Description	Competenze richieste
	9. Ripeti questo passaggio per ogni VRF che intendi estendere.	

Operazione	Description	Competenze richieste
Crea i peer Transit Gateway Connect.	1. Nell'account dell'hub di rete, segui le istruzioni in Creare un peer Transit Gateway Connect (tunnel GRE). Per questo modello, tenete presente quanto segue: • Per Name tag, inserisci un nome per il peer Transit Gateway Connect. Ti consigliamo di utilizzare un nome che corrisponda al router, ad esempio. `connectpeer-router01` • Per l'indirizzo GRE del gateway di transito, inserisci l'indirizzo IP assegnato dal blocco CIDR del gateway di transito, ad esempio. `10.100.254.1` • Per l'indirizzo PEER GRE, immettete l'indirizzo IP assegnato alla VLAN creata sul router per il VIF di transito, ad esempio. `169.254.100.1` A condizione che AWS sia in grado di raggiungere l'indirizzo IP, puoi utilizzare qualsiasi interfaccia, come VLAN o Loopback, per l'indirizzo GRE peer.	

Operazione	Description	Competenze richieste
	• Per BGP Inside CIDR Blocks (IPv4), inserisci l'indirizzo IP del blocco BGP inside CIDR, ad esempio. `169.254.101.16/29` • Per Peer ASN, inserisci l'ASN del router, ad esempio. `65534` 2. Ripeti queste istruzioni per creare un tunnel GRE per ogni router.	

Pubblicizza i percorsi verso i router

Operazione	Description	Competenze richieste
Pubblicizza i percorsi.	Associate il nuovo allegato Transit Gateway Connect alla tabella di routing creata in precedenza per questo VRF. Ad esempio, associate l'allegato Connect del gateway di transito di produzione alla tabella delle `Production-VRF` rotte. Create una route statica per il prefisso che viene pubblicizzato ai router. 1. Accedi all'account dell'hub di rete.	Amministratore di rete, architetto del cloud

Operazione	Description	Competenze richieste
	2. Apri la console Amazon VPC all'indirizzo https://console.aws.amazon.com/vpc/. 3. Nel riquadro di navigazione, in Transit Gateways, scegli Transit Gateway route tables. 4. Seleziona la tabella di instradamento del `Production-VRF` . 5. Nel menu Azioni, scegli Crea percorso statico. 6. Per CIDR, inserisci il blocco CIDR per il percorso pubblicizzato verso l'allegato del gateway di transito nel VPC di destinazione, ad esempio. `10.100.1.0/24` 7. Per Choose Attachment, scegli l'allegato Transit Gateway Connect pertinente. 8. Scegliere Create static route (Crea route statico).	

Risorse correlate

Documentazione AWS

- Documentazione Direct Connect
  - Working with Direct Connect gateways

  - Associazioni di gateway di transito
  - Interfacce virtuali AWS Direct Connect
- Documentazione Transit Gateway
  - Lavorare con i gateway di transito
  - Collegamenti del gateway di transito a un gateway Direct Connect
  - Allegati Transit Gateway Connect e peer Transit Gateway Connect
  - Creare un gateway di transito (allegato Connect)

Post sul blog di AWS

- Segmentazione delle reti ibride con AWS Transit Gateway connect
- Utilizzando AWS Transit Gateway, connettiti per estendere VRFs e aumentare la pubblicità con prefisso IP

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Ricevi notifiche Amazon SNS quando lo stato chiave di una chiave AWS KMS cambia

Shubham Harsora, Aromal Raj Jayarajan e Navdeep Pareek, Amazon Web Services

Riepilogo

I dati e i metadati associati a una chiave AWS Key Management Service (AWS KMS) vengono persi quando tale chiave viene eliminata. L'eliminazione è irreversibile e non è possibile recuperare i dati persi (compresi i dati crittografati). Puoi prevenire la perdita di dati configurando un sistema di notifica per avvisarti delle modifiche allo stato chiave delle tue chiavi AWS KMS.

Questo modello mostra come monitorare le modifiche di stato delle chiavi AWS KMS utilizzando Amazon e Amazon Simple Notification Service (EventBridge Amazon SNS) per emettere notifiche automatiche ogni volta che lo stato chiave di una chiave AWS KMS cambia in o. `Disabled` `PendingDeletion` Ad esempio, se un utente tenta di disabilitare o eliminare una chiave AWS KMS, riceverai una notifica e-mail con i dettagli sul tentativo di modifica dello stato. Puoi utilizzare questo schema anche per pianificare l'eliminazione delle chiavi AWS KMS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo con un utente AWS Identity and Access Management (IAM)
- Una chiave AWS KMS

Architecture

Stack tecnologico

- Amazon EventBridge
- AWS Key Management Service (AWS KMS)
- Amazon Simple Notification Service (Amazon SNS)

Architettura Target

Il diagramma seguente mostra un'architettura per la creazione di un processo di monitoraggio e notifica automatizzato per rilevare eventuali modifiche allo stato di una chiave AWS KMS.

Il diagramma mostra il flusso di lavoro seguente:

1. Un utente disabilita o pianifica l'eliminazione di una chiave AWS KMS.
2. Una EventBridge regola valuta il programma o l'evento`Disabled`. `PendingDeletion`
3. La EventBridge regola richiama l'argomento Amazon SNS.
4. Amazon SNS invia un messaggio di notifica e-mail agli utenti.

> **Note**
>
> Puoi personalizzare il messaggio e-mail per soddisfare le esigenze della tua organizzazione. Consigliamo di includere informazioni sulle entità in cui viene utilizzata la chiave AWS KMS. Questo può aiutare gli utenti a comprendere l'impatto dell'eliminazione della chiave AWS KMS. Puoi anche pianificare una notifica e-mail di promemoria da inviare uno o due giorni prima dell'eliminazione della chiave AWS KMS.

Automazione e scalabilità

Lo CloudFormation stack AWS distribuisce tutte le risorse e i servizi necessari per il funzionamento di questo modello. Puoi implementare il modello in modo indipendente in un singolo account o utilizzando AWS CloudFormation StackSets per più account o unità organizzative indipendenti in AWS Organizations.

Tools (Strumenti)

- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account AWS e regioni AWS. Il CloudFormation modello per questo modello descrive tutte le risorse AWS che desideri, effettua il CloudFormation provisioning e configura tali risorse per te.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. EventBridge fornisce un flusso di dati in tempo reale dalle tue applicazioni e dai servizi AWS e indirizza tali dati verso obiettivi come AWS Lambda. EventBridge semplifica il processo di creazione di architetture basate sugli eventi.

- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.

Codice

Il codice per questo modello è disponibile nell'archivio GitHub Monitor AWS KMS keys disable and scheduled delete.

Epiche

Implementa il modello CloudFormation

Operazione	Description	Competenze richieste
Clonare il repository.	Clona l'archivio GitHub Monitor AWS KMS keys, disable and scheduled delete repository sul tuo computer locale eseguendo il seguente comando: `git clone https://github.com/aws-samples/aws-kms-deletion-notification`	Amministratore AWS, architetto cloud
Aggiorna i parametri del modello.	In un editor di codice, apri il `Alerting-KMS-Events.yaml` CloudFormation modello che hai clonato dal repository, quindi aggiorna i seguenti parametri: • Ad `DestinationEmailAddress` esempio, inserisci un	Amministratore AWS, architetto cloud

Operazione	Description	Competenze richieste
	indirizzo e-mail attivo che intendi utilizzare per ricevere la notifica SNS. • Per `SNSTopicName`, inserisci un nome per il tuo argomento SNS.	
Implementa il CloudFormation modello.	1. Accedi alla console di gestione AWS e apri la console CloudFormation. 2. Nel riquadro di navigazione, scegli Crea pila, quindi scegli Con nuove risorse (standard). 3. Nella pagina Identifica risorse, scegli Avanti. 4. Nella pagina Specificare il modello, per Origine del modello, seleziona Carica un file modello. 5. Scegli file, seleziona il **`Alerting-KMS-Events.yaml`** file dal tuo GitHub repository clonato, quindi scegli Avanti. 6. Per il nome dello stack, inserisci il nome dello stack. 7. Seleziona Invia.	Amministratore AWS, architetto cloud

Confermare la sottoscrizione

Operazione	Description	Competenze richieste
Conferma l'email di iscrizione.	Dopo la corretta implement azione del CloudFormation modello, Amazon SNS invia un messaggio di conferma dell'abbonamento all'indirizzo e-mail fornito nel CloudForm ation modello. Per ricevere notifiche, devi confermare questa sottoscri zione e-mail. Per ulteriori informazioni, consulta Confermare l'abbonamento nella Amazon SNS Developer Guide.	Amministratore AWS, architett o cloud

Prova la notifica di sottoscrizione

Operazione	Description	Competenze richieste
Disabilita le chiavi AWS KMS.	1. Accedi alla Console di gestione AWS e apri la console AWS KMS. 2. Per cambiare la regione, scegli il nome della regione attualmente visualizzata, quindi scegli la regione a cui vuoi passare. 3. Nel riquadro di navigazio ne, scegli Chiavi gestite dal cliente.	Amministratore AWS

Operazione	Description	Competenze richieste
	4. Seleziona la casella di controllo per la chiave AWS KMS che desideri abilitare o disabilitare. 5. Per disabilitare la chiave AWS KMS, scegli Azioni chiave, quindi scegli Disabilita.	
Convalida l'abbonamento.	Conferma di aver ricevuto l'e-mail di notifica di Amazon SNS.	Amministratore AWS

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Elimina lo CloudFormation stack.	1. Accedi alla console di gestione AWS e apri la console CloudFormation . 2. Nel riquadro di navigazione selezionare Stacks (Stack). 3. Seleziona lo stack che hai creato in precedenza, quindi scegli Elimina.	Amministratore AWS

Risorse correlate

- AWS CloudFormation (documentazione AWS)
- Creazione di uno stack sulla CloudFormation console AWS (CloudFormation documentazione AWS)
- Creazione di architetture basate sugli eventi in AWS (documentazione di AWS Workshop Studio)
- Best practice di AWS Key Management Service (Whitepaper di AWS)

- Best practice di sicurezza per AWS Key Management Service (AWS KMS Developer Guide)

Informazioni aggiuntive

Amazon SNS fornisce la crittografia in transito per impostazione predefinita. Per allinearti alle best practice di sicurezza, puoi anche abilitare la crittografia lato server per Amazon SNS utilizzando una chiave gestita dal cliente AWS KMS.

Conserva lo spazio IP instradabile nei progetti VPC multi-account per sottoreti non destinate ai carichi di lavoro

Adam Spicer, Amazon Web Services

Riepilogo

Amazon Web Services (AWS) ha pubblicato delle best practice che consigliano l'uso di sottoreti dedicate in un cloud privato virtuale (VPC) sia per gli allegati del gateway di transito che per gli endpoint Gateway Load Balancer (per supportare AWS Network Firewall o appliance di terze parti). Queste sottoreti vengono utilizzate per contenere interfacce di rete elastiche per questi servizi. Se utilizzi sia AWS Transit Gateway che un Gateway Load Balancer, vengono create due sottoreti in ciascuna zona di disponibilità per il VPC. A causa del modo in cui VPCs sono progettate, queste sottoreti aggiuntive non possono essere più piccole di una maschera /28 e possono consumare prezioso spazio IP instradabile che potrebbe altrimenti essere utilizzato per carichi di lavoro instradabili. Questo modello dimostra come è possibile utilizzare un intervallo CIDR (Classless Inter-Domain Routing) secondario e non routabile per queste sottoreti dedicate per preservare lo spazio IP instradabile.

Prerequisiti e limitazioni

Prerequisiti

- Strategia multi-VPC per spazio IP instradabile
- Una gamma CIDR non instradabile per i servizi che stai utilizzando (allegati del gateway di transito e endpoint Gateway Load Balancer o Network Firewall)

Architecture

Architettura Target

Questo modello include due architetture di riferimento: un'architettura ha sottoreti per gli allegati Transit Gateway (TGW) e un endpoint Gateway Load Balancer (GWLBe), mentre la seconda architettura ha sottoreti solo per gli allegati TGW.

Architettura 1 – VPC collegato a TGW con routing di ingresso verso un dispositivo

Il diagramma seguente rappresenta un'architettura di riferimento per un VPC che si estende su due zone di disponibilità. In ingresso, il VPC utilizza uno schema di routing in ingresso per indirizzare il

traffico destinato alla sottorete pubblica verso un'appliance per l'ispezione del firewall. bump-in-the-wire Un allegato TGW supporta l'uscita dalle sottoreti private verso un VPC separato.

Questo modello utilizza un intervallo CIDR non instradabile per la sottorete e la sottorete degli allegati TGW. GWLBe Nella tabella di routing TGW, questo CIDR non routabile è configurato con una route blackhole (statica) utilizzando una serie di rotte più specifiche. Se le rotte dovessero essere propagate alla tabella di routing TGW, si applicherebbero queste rotte blackhole più specifiche.

In questo esempio, il CIDR instradabile /23 è suddiviso e completamente allocato a sottoreti instradabili.

Architettura 2 — VPC collegato a TGW

Il diagramma seguente rappresenta un'altra architettura di riferimento per un VPC che si estende su due zone di disponibilità. Un allegato TGW supporta il traffico in uscita (uscita) dalle sottoreti private verso un VPC separato. Utilizza un intervallo CIDR non instradabile solo per la sottorete degli allegati TGW. Nella tabella di routing TGW, questo CIDR non instradabile è configurato con una route blackhole utilizzando una serie di rotte più specifiche. Se le rotte dovessero essere propagate alla tabella di routing TGW, si applicherebbero queste rotte blackhole più specifiche.

In questo esempio, il CIDR instradabile /23 è suddiviso e completamente allocato a sottoreti instradabili.

Tools (Strumenti)

Servizi e risorse AWS

- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito. Questa rete virtuale è simile a una rete tradizionale che gestiresti nel tuo data center, con i vantaggi dell'utilizzo dell'infrastruttura scalabile di AWS. In questo modello, i VPC secondari CIDRs vengono utilizzati per preservare lo spazio IP instradabile nel carico di lavoro. CIDRs
- L'Internet Gateway Ingress Routing (associazioni edge) può essere utilizzato insieme agli endpoint Gateway Load Balancer per sottoreti dedicate non instradabili.
- AWS Transit Gateway è un hub centrale che collega reti locali VPCs e internazionali. In questo modello, VPCs sono collegati centralmente a un gateway di transito e gli allegati del gateway di transito si trovano in una sottorete dedicata non instradabile.

- Gateway Load Balancer: consentono di implementare, dimensionare e gestire appliance virtuali, come firewall, sistemi di prevenzione e rilevamento delle intrusioni e sistemi di ispezione approfondita dei pacchetti. Il gateway funge da unico punto di ingresso e uscita per tutto il traffico. In questo modello, gli endpoint per un Gateway Load Balancer possono essere utilizzati in una sottorete dedicata non routabile.
- AWS Network Firewall è un firewall di rete a stato gestito e un servizio di rilevamento e prevenzione delle intrusioni VPCs nel cloud AWS. In questo modello, gli endpoint di un firewall possono essere utilizzati in una sottorete dedicata non routabile.

Archivio di codice

Un runbook e CloudFormation modelli AWS per questo pattern sono disponibili nel repository GitHub Non-Routable Secondary CIDR Patterns. Puoi utilizzare i file di esempio per configurare un laboratorio di lavoro nel tuo ambiente.

Best practice

AWS Transit Gateway

- Utilizza una sottorete separata per ogni allegato VPC del gateway di transito.
- Alloca una sottorete /28 dall'intervallo CIDR secondario non routabile per le sottoreti allegate del gateway di transito.
- In ogni tabella di routing del gateway di transito, aggiungi una route statica e più specifica per l'intervallo CIDR non instradabile come buco nero.

Gateway Load Balancer e routing in ingresso

- Utilizza il routing in ingresso per indirizzare il traffico da Internet agli endpoint Gateway Load Balancer.
- Utilizza una sottorete separata per ogni endpoint Gateway Load Balancer.
- Alloca una sottorete /28 dall'intervallo CIDR secondario non instradabile per le sottoreti degli endpoint Gateway Load Balancer.

Epiche

Crea VPCs

Operazione	Description	Competenze richieste
Determina l'intervallo CIDR non instradabile.	Determina un intervallo CIDR non instradabile che verrà utilizzato per la sottorete degli allegati del gateway di transito e (facoltativamente) per qualsiasi sottorete endpoint Gateway Load Balancer o Network Firewall. Questo intervallo CIDR verrà utilizzato come CIDR secondario per il VPC. Non deve essere instradabile dall'intervallo CIDR primario del VPC o dalla rete più ampia.	Architetto del cloud
Determina gli intervalli CIDR instradabili per. VPCs	Determina un set di intervalli CIDR instradabili che verranno utilizzati per il tuo. VPCs Questo intervallo CIDR verrà utilizzato come CIDR principale per il tuo. VPCs	Architetto del cloud
Crea VPCs.	Crea i tuoi VPCs e collegali al gateway di transito. Ogni VPC deve avere un intervallo CIDR primario instradabile e un intervallo CIDR secondario non instradabile, in base agli intervalli determinati nei due passaggi precedenti.	Architetto del cloud

Configura i percorsi blackhole del Transit Gateway

Operazione	Description	Competenze richieste
Crea buchi neri più specifici e non CIDRs instradabili.	Ogni tabella di routing del gateway di transito deve disporre di una serie di percorsi blackhole creati per i percorsi non instradabili. CIDRs Questi sono configura ti per garantire che il traffico proveniente dal CIDR VPC secondario rimanga non instradabile e non si disperda nella rete più grande. Questi percorsi devono essere più specifici del CIDR non routabile impostato come CIDR secondario sul VPC. Ad esempio, se il CIDR secondari o non routabile è 100.64.0. 0/26, le rotte dei buchi neri nella tabella di routing del gateway di transito devono essere 100.64.0.0/27 e 100.64.0.32/27.	Architetto del cloud

Risorse correlate

- Best practices for deploying Gateway Load Balancer
- Architetture di ispezione distribuite con Gateway Load Balancer
- Giornata di immersione nella rete – Laboratorio firewall da Internet a VPC
- Transit gateway design best practices

Informazioni aggiuntive

La gamma CIDR secondaria non routabile può essere utile anche quando si lavora con implementazioni di container su larga scala che richiedono un ampio set di indirizzi IP. È possibile utilizzare questo modello con un gateway NAT privato per utilizzare una sottorete non instradabile per ospitare le distribuzioni dei contenitori. Per ulteriori informazioni, consulta il post del blog Come risolvere l'esaurimento dell'IP privato con la soluzione NAT privata.

Fornisci un prodotto Terraform AWS Service Catalog utilizzando un repository di codice

Dott. Rahul Sharad Gaikwad e Tamilselvan P, Amazon Web Services

Riepilogo

AWS Service Catalog supporta il provisioning self-service con governance per le configurazioni Terraform. HashiCorp Se utilizzi Terraform, puoi utilizzare Service Catalog come strumento unico per organizzare, governare e distribuire le tue configurazioni Terraform su larga scala. AWS È possibile accedere alle funzionalità principali di Service Catalog, tra cui la catalogazione di modelli di infrastruttura come codice (IaC) standardizzati e preapprovati, il controllo degli accessi, il provisioning delle risorse cloud con accesso con privilegi minimi, il controllo delle versioni, la condivisione con migliaia di utenti e il tagging. Account AWS Gli utenti finali, come ingegneri, amministratori di database e data scientist, visualizzano un elenco di prodotti e versioni a cui hanno accesso e possono implementarli con una singola azione.

Questo modello ti aiuta a distribuire AWS le risorse utilizzando il codice Terraform. È possibile accedere al codice Terraform nel GitHub repository tramite Service Catalog. Utilizzando questo approccio, integri i prodotti con i flussi di lavoro Terraform esistenti. Gli amministratori possono creare portafogli Service Catalog e aggiungervi AWS Launch Wizard prodotti utilizzando Terraform.

Di seguito sono riportati i vantaggi di questa soluzione:

- Grazie alla funzionalità di rollback di Service Catalog, in caso di problemi durante la distribuzione, è possibile ripristinare il prodotto a una versione precedente.
- È possibile identificare facilmente le differenze tra le versioni del prodotto. Ciò consente di risolvere i problemi durante la distribuzione.
- È possibile configurare una connessione al repository in Service Catalog, ad esempio a GitHub o GitLab. È possibile apportare modifiche al prodotto direttamente tramite il repository.

Per informazioni sui vantaggi complessivi di AWS Service Catalog, vedere What is Service Catalog.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS.

- A GitHub BitBucket, o altro repository che contiene i file di configurazione Terraform in formato ZIP.
- AWS Serverless Application Model Interfaccia a riga di comando (AWS SAM CLI), installata.
- AWS Command Line Interface (AWS CLI), installato e configurato.
- Vai, installato.
- Python versione 3.9, installata. AWS SAM CLI richiede questa versione di Python.
- Autorizzazioni per scrivere ed eseguire AWS Lambda funzioni e autorizzazioni per accedere e gestire i prodotti e i portafogli Service Catalog.

Architecture

Il diagramma mostra il flusso di lavoro seguente:

1. Quando una configurazione Terraform è pronta, uno sviluppatore crea un file.zip che contiene tutto il codice Terraform. Lo sviluppatore carica il file.zip nell'archivio di codice collegato a Service Catalog.
2. Un amministratore associa il prodotto Terraform a un portafoglio in Service Catalog. L'amministratore crea inoltre un vincolo di avvio che consente agli utenti finali di fornire il prodotto.
3. In Service Catalog, gli utenti finali avviano AWS le risorse utilizzando la configurazione Terraform. Possono scegliere quale versione del prodotto distribuire.

Tools (Strumenti)

Servizi AWS

- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS Service Catalogti aiuta a gestire centralmente i cataloghi dei servizi IT approvati. AWS Gli utenti finali possono distribuire rapidamente soltanto i servizi IT approvati di cui hanno bisogno, in accordo con i vincoli stabiliti dall'organizzazione.

Altri servizi

- Go è un linguaggio di programmazione open source supportato da Google.

- Python è un linguaggio di programmazione per computer generico.

Deposito di codice

Se hai bisogno di configurazioni Terraform di esempio da distribuire tramite Service Catalog, puoi utilizzare le configurazioni nell'Amazon Macie GitHub Organization Setup Using Terraform repository. L'uso degli esempi di codice in questo repository non è richiesto.

Best practice

- Invece di fornire i valori per le variabili nel file di configurazione Terraform (`terraform.tfvars`), configura i valori delle variabili quando avvii il prodotto tramite Service Catalog.
- Concedi l'accesso al portafoglio solo a utenti o amministratori specifici.
- Segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione AWS Identity and Access Management (IAM).

Epiche

Configura la tua workstation locale

Operazione	Description	Competenze richieste
(Facoltativo) Installa Docker.	Se desideri eseguire AWS Lambda le funzioni nel tuo ambiente di sviluppo, installa Docker. Per ulteriori informazioni, consulta la sezione Installazione del motore Docker nella documentazione di Docker.	DevOps ingegnere
Installa il AWS Service Catalog motore per Terraform.	1. Immettere il seguente comando per clonare il repository AWS Service Catalog Engine for Terraform.	DevOps ingegnere, amministratore AWS

Operazione	Description	Competenze richieste

	`git clone https://github.com/aws-samples/service-catalog-engine-for-terraform-os.git`
	2. Passa alla directory principale del repository clonato.
	3. Inserire il seguente comando. Questo installa il motore.
	`run ./bin/bash/deploy-tre.sh -r`
	Il valore Regione AWS impostato nel profilo predefinito non viene utilizzato durante l'install azione automatizzata. Al contrario, si fornisce la regione quando si esegue questo comando.	

Connect il GitHub repository

Operazione	Description	Competenze richieste
Crea una connessione al GitHub repository.	1. Accedi a Console di gestione AWS, quindi apri la console Developer Tools. Puoi accedere alla console Developer Tools scegliend	Amministratore AWS

Operazione	Description	Competenze richieste
	o un servizio come AWS CodePipeline o AWS CodeDeploy. 2. Nel riquadro di navigazione a sinistra, scegli Impostazioni, quindi scegli Connessioni. 3. Scegli Crea connessione. 4. Seleziona il repository in cui conservi il codice sorgente Terraform. Ad esempio, puoi scegliere Bitbucket o Enterprise Server GitHub. GitHub 5. Inserisci un nome per la connessione, quindi scegli Connetti. 6. Quando richiesto, autentica il repository. Una volta completata l'autenticazione, la connessione viene creata e lo stato diventa attivo.	

Crea un prodotto Terraform in Service Catalog

Operazione	Description	Competenze richieste
Crea il prodotto Service Catalog.	1. Apri la AWS Service Catalog console.	Amministratore AWS

Operazione	Description	Competenze richieste
	2. Vai alla sezione Amministrazione, quindi scegli Elenco prodotti. 3. Scegli Crea prodotto. 4. Nella pagina Crea prodotto nella sezione Dettagli del prodotto, scegli il Tipo di prodotto esterno. Service Catalog utilizza questo tipo di prodotto per supportare i prodotti Terraform Community Edition. 5. Inserisci un nome e un proprietario per il prodotto Service Catalog. 6. Seleziona Specificare il repository di codice utilizzando un CodeStar provider. 7. Inserisci le seguenti informazioni per il tuo repository: • Connettiti al tuo provider utilizzando AWS CodeConnections: seleziona la connessione che hai creato in precedenza. • Repository: seleziona il repository. • Ramo: seleziona il ramo. • Percorso del file modello: scegli il percorso in cui è archiviato il file	

Operazione	Description	Competenze richieste
	del modello di codice. Il nome del file deve terminare con `tar.gz`. 8. In Nome e descrizione della versione, fornisci informazioni sulla versione del prodotto. 9. Scegli Crea prodotto.	
Crea un portfolio.	1. Apri la AWS Service Catalog console. 2. Vai alla sezione Amministrazione, quindi scegli Portafogli. 3. Scegli Crea portfolio. 4. Immetti uno dei seguenti valori: • Portfolio name (Nome portafoglio) - `Sample terraform` • Descrizione del portafoglio: `Sample portfolio for Terraform configurations` • Proprietario: le tue informazioni di contatto, ad esempio un indirizzo e-mail 5. Scegli Create (Crea).	Amministratore AWS

Operazione	Description	Competenze richieste
Aggiungi il prodotto Terraform al portafoglio.	1. Apri la AWS Service Catalog console. 2. Vai alla sezione Amministrazione, quindi scegli Elenco prodotti. 3. Seleziona il prodotto Terraform che hai creato in precedenza. 4. Scegli Azioni, quindi scegli Aggiungi prodotto al portafoglio. 5. Scegli il `Sample terraform` portfolio. 6. Scegli Aggiungi prodotto al portafoglio.	Amministratore AWS

Operazione	Description	Competenze richieste
Creare la policy d'accesso.	1. Apri la console AWS Identity and Access Management (IAM). 2. Nel pannello di navigazione, seleziona Policies (Policy). 3. Nel riquadro del contenuto seleziona Create policy (Crea policy). 4. Scegli l'opzione JSON. 5. Inserisci la policy JSON di esempio nella policy di accesso nella sezione Informazioni aggiuntive di questo modello. 6. Scegli Next (Successivo). 7. Nella pagina Rivedi e crea, nella casella Nome della policy, inserisci `TerraformResourceCreationAndArtifactAccessPolicy` . 8. Scegli Crea policy.	Amministratore AWS

Operazione	Description	Competenze richieste
Crea una politica di fiducia personalizzata.	1. Aprire la console IAM. 2. Nel pannello di navigazione, selezionare Ruoli. 3. Selezionare Create role (Crea ruolo). 4. In Tipo di entità affidabile, scegli Politica di fiducia personalizzata. 5. Nell'editor di policy JSON, inserisci la policy JSON di esempio in Trust policy nella sezione Informazioni aggiuntive di questo modello. 6. Scegli Next (Successivo). 7. In Politiche di autorizzazione, scegli `TerraformResourceCreationAndArtifactAccessPolicy` quella che hai creato in precedenza. 8. Scegli Next (Successivo). 9. In Dettagli del ruolo, nella casella Nome ruolo, immettere `SCLaunch-product` . **Important** Il nome del ruolo deve iniziare con `SCLaunch`.	Amministratore AWS

Operazione	Description	Competenze richieste
	10. Scegli Crea ruolo.	
Aggiungi un vincolo di lancio al prodotto Service Catalog.	1. Accedi Console di gestione AWS come utente con autorizzazioni amministrative. 2. Apri la AWS Service Catalog console. 3. Nel riquadro di navigazione, scegli Portafogli. 4. Scegli il portfolio che hai creato in precedenza. 5. Nella pagina dei dettagli del portfolio, scegli la scheda Vincoli, quindi scegli Crea vincolo. 6. Per Prodotto, seleziona il prodotto Terraform che hai creato in precedenza. 7. In Vincolo di avvio, per Metodo, scegli Inserisci il nome del ruolo. 8. Nella casella Nome del ruolo, immettete. `SCLaunch-product` 9. Scegli Create (Crea).	Amministratore AWS

Operazione	Description	Competenze richieste
Concedi l'accesso al prodotto.	1. Apri la AWS Service Catalog console. 2. Nel riquadro di navigazione, scegli Portafogli. 3. Scegli il portfolio che hai creato in precedenza. 4. Scegli la scheda Accesso, quindi scegli Concedi l'accesso. 5. Scegli la scheda Ruoli, quindi seleziona il ruolo a cui devi avere accesso per distribuire questo prodotto. 6. Selezionare Concedi l'accesso.	Amministratore AWS
Avvia il prodotto.	1. Accedi Console di gestione AWS come utente con le autorizzazioni per distribuire il prodotto Service Catalog. 2. Apri la AWS Service Catalog console. 3. Nel riquadro di navigazione, scegli Prodotti. 4. Scegli il prodotto che hai creato in precedenza, quindi scegli Avvia prodotto. 5. Inserisci il nome di un prodotto e definisci i parametri richiesti. 6. Scegli Launch product.	DevOps ingegnere

Verifica della distribuzione

Operazione	Description	Competenze richieste
Convalida la distribuzione.	Esistono due macchine a AWS Step Functions stati per il flusso di lavoro di provisioning del Service Catalog: • `ManageProvisionedProductStateMachine` — Service Catalog richiama questa macchina a stati durante il provisioning di un nuovo prodotto Terraform e durante l'aggiornamento di un prodotto fornito Terraform esistente. • `TerminateProvisionedProductStateMachine` — Service Catalog richiama questa macchina a stati quando termina un prodotto fornito da Terraform esistente. Si controllano i registri della macchina a stati per confermare che il prodotto è `ManageProvisionedProductStateMachine` stato fornito. 1. Accedere a Console di gestione AWS, quindi aprire	DevOps ingegnere

Operazione	Description	Competenze richieste
	la AWS Step Functions console. 2. Nel riquadro di navigazione a sinistra, scegli Macchine a stati. 3. Sceglietе`ManagePro visionedProductSta teMachine` . 4. Nell'elenco Esecuzioni, inserite l'ID del prodotto assegnato per individuare l'esecuzione. **Note** I nomi dei bucket di backend del file di stato iniziano con. `sc-terraform-engine-state-` 5. Verifica che tutte le risorse richieste siano state create nell'account.	

Pulisci l'infrastruttura

Operazione	Description	Competenze richieste
Eliminare i prodotti forniti.	1. Accedi Console di gestione AWS come utente con le autorizzazioni per distribuire il prodotto Service Catalog.	DevOps ingegnere

Operazione	Description	Competenze richieste
	2. Apri la AWS Service Catalog console. 3. Nella barra di navigazione a sinistra, scegli Provisioned products. 4. Seleziona il prodotto che hai creato. 5. Nell'elenco Azioni, scegli Termina. 6. Nella casella di testo di conferma, inserisci `terminate` , quindi scegli Termina il prodotto fornito. 7. Ripeti questi passaggi per terminare tutti i prodotti forniti.	

Operazione	Description	Competenze richieste
Rimuovi il AWS Service Catalog motore per Terraform.	1. Accedi Console di gestione AWS come utente con autorizzazioni amministrative. 2. Apri la console Amazon Simple Storage Service (Amazon S3). 3. Nel pannello di navigazione, scegli Bucket. 4. Seleziona il `sc-terraform-engine-logging-XXXX` bucket. 5. Scegli Vuoto. 6. Ripeti i passaggi da 4 a 5 per i seguenti bucket: • `sc-terraform-engine-state-XXXX` • `terraform-engine-bootstrap-XXXX` 7. Apri la AWS CloudFormation console, quindi verifica di essere nella posizione corretta. Regione AWS 8. Nella barra di navigazione a sinistra, scegli Stacks. 9. Seleziona`SAM-TRE`, quindi scegli Elimina. Attendi che lo stack sia stato eliminato. 10. Seleziona`Bootstrap-TRE`, quindi scegli Elimina.	Amministratore AWS

Operazione	Description	Competenze richieste
	Attendi che lo stack sia stato eliminato.	

Risorse correlate

AWS documentazione

- Iniziare con un prodotto Terraform

Documentazione Terraform

- Installazione di Terraform
- Configurazione del backend Terraform
- Documentazione Terraform Provider AWS

Informazioni aggiuntive

Politica di accesso

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "s3:GetObject",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "s3:ExistingObjectTag/servicecatalog:provisioning": "true"
                }
            }
        },
        {
            "Action": [
                "s3:CreateBucket*",
                "s3:DeleteBucket*",
```

```
                "s3:Get*",
                "s3:List*",
                "s3:PutBucketTagging"
            ],
            "Resource": "arn:aws:s3:::*",
            "Effect": "Allow"
        },
        {
            "Action": [
                "resource-groups:CreateGroup",
                "resource-groups:ListGroupResources",
                "resource-groups:DeleteGroup",
                "resource-groups:Tag"
            ],
            "Resource": "*",
            "Effect": "Allow"
        },
        {
            "Action": [
                "tag:GetResources",
                "tag:GetTagKeys",
                "tag:GetTagValues",
                "tag:TagResources",
                "tag:UntagResources"
            ],
            "Resource": "*",
            "Effect": "Allow"
        }
    ]
}
```

Policy di trust

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "GivePermissionsToServiceCatalog",
            "Effect": "Allow",
            "Principal": {
                "Service": "servicecatalog.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
```

```
        },
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::account_id:root"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringLike": {
                    "aws:PrincipalArn": [
                        "arn:aws:iam::accounti_id:role/TerraformEngine/
TerraformExecutionRole*",
                        "arn:aws:iam::accounti_id:role/TerraformEngine/
ServiceCatalogExternalParameterParserRole*",
                        "arn:aws:iam::accounti_id:role/TerraformEngine/
ServiceCatalogTerraformOSParameterParserRole*"
                    ]
                }
            }
        }
    ]
}
```

Registrazione di più indirizzi e-mail Account AWS con un unico indirizzo e-mail utilizzando Amazon SES

Joe Wozniak e Shubhangi Vishwakarma, Amazon Web Services

Riepilogo

Questo schema descrive come è possibile disaccoppiare gli indirizzi e-mail reali dall'indirizzo e-mail associato a un. Account AWS Account AWS richiede che venga fornito un indirizzo email univoco al momento della creazione dell'account. In alcune organizzazioni, il team che gestisce Account AWS deve assumersi l'onere di gestire molti indirizzi e-mail univoci con il proprio team di messaggistica. Questo può essere difficile per le grandi organizzazioni che ne gestiscono molti Account AWS. Inoltre, se il sistema di posta elettronica non consente l'indirizzamento positivo o secondario, come definito in Sieve Email Filtering: Subaddress Extension (RFC 5233), aggiungendo un segno più (+) e un identificatore alla fine della parte locale dell'indirizzo e-mail, ad esempio, questo schema può aiutare a superare questa limitazione. `admin+123456789123@example.com`

Questo modello fornisce una soluzione di vendita di indirizzi e-mail unica che consente ai proprietari di associare un indirizzo e-mail a più indirizzi e-mail. Account AWS Account AWS Gli indirizzi e-mail reali dei Account AWS proprietari vengono quindi associati a questi indirizzi e-mail generati in una tabella. La soluzione gestisce tutte le e-mail in arrivo per gli account e-mail univoci, cerca il proprietario di ciascun account e quindi inoltra i messaggi ricevuti al proprietario.

Prerequisiti e limitazioni

Prerequisiti

- Accesso amministrativo a un. Account AWS
- Accesso a un ambiente di sviluppo.
- (Facoltativo) La familiarità con AWS Cloud Development Kit (AWS CDK) i flussi di lavoro e il linguaggio di programmazione Python ti aiuterà a risolvere eventuali problemi o apportare modifiche.

Limitazioni

- Lunghezza complessiva dell'indirizzo e-mail fornito di 64 caratteri. Per i dettagli, CreateAccountconsulta il riferimento all'AWS Organizations API.

Versioni del prodotto

- Node.js versione 22.x o successiva
- Python 3.13 o successivo
- Pacchetti Python pip e virtualenv
- AWS CDK CLI versione 2.1019.2 o successiva
- Docker 20.10.x o versione successiva

Architecture

Stack tecnologico Target

- CloudFormation pila
- AWS Lambda funzioni
- Regola e set di regole di Amazon Simple Email Service (Amazon SES)
- AWS Identity and Access Management Ruoli e politiche (IAM)
- Politica relativa ai bucket e ai bucket di Amazon Simple Storage Service (Amazon S3)
- AWS Key Management Service (AWS KMS) chiave e politica chiave
- Argomento e policy tematica di Amazon Simple Notification Service (Amazon SNS)
- Tabella Amazon DynamoDB

Architettura Target

Questo diagramma mostra due flussi:

- Flusso di vendita degli indirizzi e-mail: nel diagramma, il flusso di vendita degli indirizzi e-mail (sezione inferiore) inizia in genere con una soluzione di vendita di account o un'automazione esterna, oppure viene richiamato manualmente. Nella richiesta, viene chiamata una funzione Lambda con un payload che contiene i metadati necessari. La funzione utilizza queste informazioni per generare un nome account e un indirizzo e-mail univoci, li archivia in un database DynamoDB e restituisce i valori al chiamante. Questi valori possono quindi essere utilizzati per crearne uno nuovo Account AWS (in genere utilizzando). AWS Organizations
- Flusso di inoltro delle e-mail: questo flusso è illustrato nella sezione superiore del diagramma precedente. Quando Account AWS viene creato un indirizzo e-mail utilizzando l'account, l'e-mail

generata dal flusso di vendita degli indirizzi e-mail, AWS invia diverse e-mail, come la conferma della registrazione dell'account e le notifiche periodiche, a tale indirizzo e-mail. Seguendo i passaggi indicati in questo schema, Account AWS configuri Amazon SES per ricevere e-mail per l'intero dominio. Questa soluzione configura regole di inoltro che consentono a Lambda di elaborare tutte le e-mail in arrivo, verificare se l'indirizzo T0 è nella tabella DynamoDB e inoltrare invece il messaggio all'indirizzo e-mail del proprietario dell'account. L'utilizzo di questo processo offre ai proprietari degli account la possibilità di associare più account a un unico indirizzo e-mail.

Automazione e scalabilità

Questo modello utilizza il AWS CDK per automatizzare completamente l'implementazione.
La soluzione utilizza servizi AWS gestiti che scaleranno automaticamente (o possono essere configurati per) soddisfare le vostre esigenze. Le funzioni Lambda potrebbero richiedere una configurazione aggiuntiva per soddisfare le tue esigenze di scalabilità. Per ulteriori informazioni, consulta Understanding Lambda Function Scaling nella documentazione di Lambda.

Tools (Strumenti)

Servizi AWS

- CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita in tutte le regioni. Account AWS
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella shell della riga di comando.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Simple Email Service (Amazon SES) Simple Email Service (Amazon SES) ti aiuta a inviare e ricevere e-mail utilizzando i tuoi indirizzi e-mail e domini.

- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Strumenti necessari per la distribuzione

- Ambiente di sviluppo con accesso IAM al tuo Account AWS. AWS CLI Per i dettagli, consulta i collegamenti nella sezione Risorse correlate.
- Sul tuo sistema di sviluppo, installa quanto segue:
  - Strumento da riga di comando Git, disponibile dal sito Web Git downloads.
  - AWS CLI Per configurare le credenziali di accesso per. AWS CDK Per ulteriori informazioni, consulta la documentazione relativa ad AWS CLI.
  - Python versione 3.13 o successiva, disponibile dal sito Web di download di Python.
  - UV per la gestione dei pacchetti in Python. Per le istruzioni di installazione, consulta la guida all'installazione UV.
  - Node.js versione 22.x o successiva. Per le istruzioni di installazione, consulta la documentazione di Node.js.
  - AWS CDK CLI versione 2.1019.2 o successiva. Per le istruzioni di installazione, consulta la documentazione.AWS CDK
  - Docker versione 20.10.x o successiva. Per le istruzioni di installazione, consulta la documentazione di Docker.

Codice

Il codice per questo pattern è disponibile nell'archivio di posta elettronica di GitHub Account AWS fabbrica.

Epiche

Assegna un ambiente di implementazione target

Operazione	Description	Competenze richieste
Identifica o crea un Account AWS.	Identifica una soluzione di posta elettronica esistente o	Amministratore AWS, amministratore cloud

Operazione	Description	Competenze richieste
	nuova Account AWS a cui hai pieno accesso amministrativo per implementare la soluzione di posta elettronica.	

Operazione	Description	Competenze richieste
Configura un ambiente di distribuzione.	Configura un ambiente di distribuzione facile da usare e imposta le dipendenze seguendo questi passaggi: 1. Configura il tuo ambiente di sviluppo con gli strumenti elencati nella sezione Strumenti. 2. Clona il codice base del repository di posta elettronica di GitHub Account AWS fabbrica nel tuo ambiente di sviluppo utilizzando il comando: `git clone https://github.com/aws-samples/aws-account-factory-email` 3. Nel `requirements.in` file (nella radice del repository), aggiorna la riga che inizia con in modo che `aws-cdk-lib==` corrisponda alla versione più recente di. AWS CDK Per identificare qual è la versione più recente, esegui `npm view aws-cdk-lib` e cerca la parola chiave «latest:». 4. Se hai effettuato un aggiornamento alla	AWS DevOps, sviluppatore di app

Operazione	Description	Competenze richieste
	versione in `requirements.in` , esegui il seguente comando per aggiornare il `requirements.txt` file utilizzando UV: `uv pip compile requirements.in -o requirements.txt --python-version 3.13`	

Configura un dominio verificato

Operazione	Description	Competenze richieste
Identifica e assegna un dominio.	La funzionalità di inoltro delle e-mail richiede un dominio dedicato. Identifica e assegna un dominio o sottodominio che puoi verificare con Amazon SES. Questo dominio dovrebbe essere disponibile per ricevere e-mail in arrivo all'interno del Account AWS luogo in cui è implementata la soluzione di inoltro e-mail. Requisiti del dominio: • Il dominio deve essere un dominio o un sottodominio standard. • Il dominio deve essere risolvibile tramite DNS	Amministratore cloud, amministratore di rete, amministratore DNS

Operazione	Description	Competenze richieste
	esternamente, poiché verrà utilizzato per ricevere e-mail dall'esterno dell'orga nizzazione.	
Verifica il dominio.	Verifica che il dominio identific ato possa essere utilizzato per accettare la posta elettronica in arrivo. Completa le istruzioni in Verifica del tuo dominio per la ricezione di e-mail di Amazon SES nella documentazione di Amazon SES. Ciò richiederà il coordinamento con la persona o il team responsabile dei record DNS del dominio.	Sviluppatore di app, AWS DevOps
Configura i record MX.	Configura il tuo dominio con record MX che puntano agli endpoint Amazon SES nella tua regione Account AWS e. Per ulteriori informazioni, consulta Pubblicazione di un record MX per la ricezione di e-mail di Amazon SES nella documentazione di Amazon SES.	Amministratore cloud, amministratore di rete, amministratore DNS

Implementa la soluzione di vendita e inoltro della posta elettronica

Operazione	Description	Competenze richieste
Modificare i valori predefiniti in`cdk.json`.	Modifica alcuni dei valori predefiniti nel `cdk.json` file (nella radice del repository) in modo che la soluzione funzioni correttamente dopo la distribuzione. 1. Modifica il `SES_DOMAIN_NAME` valore in modo che corrisponda al nome di dominio verificato in precedenza. 2. Modifica il `ADDRESS_FROM` valore per includere lo stesso dominio in cui si trova`SES_DOMAIN_NAME` . La parte locale dell'indirizzo deve essere determinata dal tuo team cloud. Questo indirizzo diventa l'`FROM`indirizzo di ogni e-mail inoltrata tramite la soluzione. 3. Modifica il `ADDRESS_ADMIN` valore in modo che corrisponda all'indirizzo e-mail a cui verranno inoltrati tutti i messaggi in arrivo non corrispondenti. Questo valore deve essere un indirizzo e-mail valido e operativo.	Sviluppatore di app, AWS DevOps

Operazione	Description	Competenze richieste

Implementa la soluzione di vendita e inoltro della posta elettronica.	1. Crea un ambiente virtuale Python:
	`uv venv --python 3.13`
	2. Attiva l'ambiente virtuale Python:
	`source .venv/bin/activate`
	Oppure, sulla piattaforma Windows, usa:
	`% .venv\Scripts\activate.bat`
	3. Installa tutti i requisiti di Python senza errori:
	`uv pip install -r requirements.txt`
	4. Sintetizza il modello: CloudFormation
	`cdk synth`
	Verifica che non vi siano errori e che il CloudFormation modello completo contenga l'output previsto.
	5. (Facoltativo) Se state distribuendo il AWS CDK codice nella regione corrente Account AWS o
		Sviluppatore di app, AWS DevOps

Operazione	Description	Competenze richieste

	nella regione per la prima volta, avviate l'ambient e. Per ulteriori informazi oni, consulta AWS CDK Bootstrap nella documenta zione. AWS CDK
	`cdk bootstrap aws://AWS-ACCOUNT-NUMBER/REGION`
	Sostituisci `AWS-ACCOUNT-NUMBER` e `REGION` con valori effettivi.
	6. Implementa la soluzione:
	`cdk bootstrap`
	`cdk deploy`
	I comandi devono essere completati senza errori.	

Operazione	Description	Competenze richieste
Verifica che la soluzione sia stata implementata.	Verifica che la soluzione sia stata implementata correttamente prima di iniziare i test: 1. Apri la CloudFormation console e cerca uno CloudFormation stack che contenga il nome. `AwsMailFwdStack` 2. Verifica che questo `AwsMailFwdStack` stack contenga le seguenti risorse: • Funzioni Lambda • Regola e set di regole di Amazon SES • Ruoli IAM e policy • Politica relativa ai bucket e ai bucket di Amazon S3 • AWS KMS chiave e politica chiave • Argomento e policy tematica di Amazon SNS • DynamoDB tabella	Sviluppatore di app, AWS DevOps

Verifica che la distribuzione e l'inoltro delle e-mail funzionino come previsto

Operazione	Description	Competenze richieste
Verifica che l'API funzioni.	In questo passaggio, invii i dati di test all'API della soluzione e confermi che la soluzione produca l'output previsto e che	Sviluppatore di app, AWS DevOps

Operazione	Description	Competenze richieste
	le operazioni di backend siano state eseguite come previsto. Esegui manualmente la funzione Vend Email Lambda utilizzando l'input di test. (Per un esempio, vedete il file sample_vend_request.json). Per, utilizza un indirizzo email valido. `OwnerAddress` L'API dovrebbe restituire il nome dell'account e l'indirizzo e-mail dell'account con i valori previsti.	

Operazione	Description	Competenze richieste
Verifica che l'email venga inoltrata.	In questo passaggio, invii un'e-mail di prova tramite il sistema e verifichi che l'e-mail venga inoltrata al destinatario previsto. 1. Ottieni l'e-mail dell'account dall'ultimo passaggio. 2. Invia un'e-mail a questo indirizzo con l'oggetto del test e il corpo del testo. 3. Conferma di aver ricevuto l'e-mail all'indirizzo e-mail del proprietario dell'account. 4. Verifica che l'e-mail che hai ricevuto abbia un `FROM` indirizzo che corrisponde all'`ADDRESS_FROM` impostazione `incdk.json`. 5. Verifica che l'oggetto e il corpo dell'email ricevuta coincidano con il messaggio originale inviato.	Sviluppatore di app, AWS DevOps

risoluzione dei problemi

Problema	Soluzione
Il sistema non inoltra le e-mail come previsto.	Verifica che la configurazione sia corretta: 1. Dovresti aver completato la procedura di verifica Amazon SES per il tuo dominio.

Problema	Soluzione
	2. Il tuo dominio deve essere configurato correttamente con record MX che puntano agli endpoint Amazon SES nella tua regione Account AWS e nella tua regione. Per ulteriori informazioni, consulta Pubblicazione di un record MX per la ricezione di e-mail di Amazon SES nella documentazione di Amazon SES. Dopo aver verificato la configurazione del dominio, segui questi passaggi: 1. Apri la CloudWatch console Amazon per l'account e la regione in cui hai distribuito la soluzione e vai ai gruppi di CloudWatch log nel riquadro di navigazione. 2. Cerca nell'elenco dei gruppi di log per `SesMailForwardLogGroup` . 3. Esamina i log di questo gruppo per vedere se vengono generati errori durante il processo di vendita e inoltro delle e-mail.

Problema	Soluzione
Quando tenti di distribuire lo AWS CDK stack, ricevi un errore simile a: «Errore di formato del modello: tipi di risorse non riconosciuti»	Nella maggior parte dei casi, questo messaggio di errore indica che la regione a cui ti rivolgi non dispone di tutti i servizi AWS disponibili. Se utilizzi un' EC2 istanza Amazon per distribuire la soluzione, potresti scegliere come target una regione diversa da quella in cui è in esecuzione l'istanza. **Note** Per impostazione predefinita, AWS CDK viene distribuito nella regione e nell'account configurati in. AWS CLI Possibili soluzioni: 1. Verifica se tutti i servizi necessari per questa soluzione (consulta la sezione relativa allo stack tecnologico Target all'inizio di questo modello) rientrano tra quelli a Regione AWS cui ti rivolgi esaminando i servizi AWS per regione. 2. Se utilizzi un' EC2 istanza e hai come target una regione diversa da quella in cui è in esecuzione, assicurati di impostare la variabile di `AWS_DEFAULT_REGION` ambiente o di impostare una regione con la AWS CLI prima di distribuire la soluzione. Per ulteriori informazioni, consulta la sezione Configurazione delle variabili di ambiente AWS CLI nella documentazione. AWS CLI In alternativa, puoi modificare il `app.py` file nella radice del repository per includere un ID account e una regione codificati seguendo

Problema	Soluzione
	le istruzioni nella documentazione per gli ambienti.AWS CDK
Quando si distribuisce la soluzione, viene visualizzato il messaggio di errore: «Distribuzione non riuscita: Errore: AwsMailFwdStack: parametro SSM bootstrap/hnb659fds/version /cdk- non trovato. L'ambiente è stato avviato? Per favore esegui 'cdk bootstrap'»	Se non hai mai distribuito AWS CDK risorse nella regione Account AWS and che hai scelto come target, dovrai prima eseguire il comando come indicato dall'errore. `cdk bootstrap` Se continui a ricevere questo errore dopo aver eseguito il comando bootstrap, potresti provare a distribuire la soluzione in una regione diversa da quella in cui è in esecuzione il tuo ambiente di sviluppo. Per risolvere questo problema, imposta la variabile di `AWS_DEFAULT_REGION` ambiente o imposta una regione con la AWS CLI prima di distribuire la soluzione. In alternativa, puoi modificare il `app.py` file nella radice del repository per includere un ID account e una regione codificati seguendo le istruzioni nella documentazione per gli ambienti.AWS CDK

Risorse correlate

- Per informazioni sull'installazione di AWS CLI, consulta Installazione o aggiornamento alla versione più recente di. AWS CLI
- Per informazioni sulla configurazione delle credenziali di accesso AWS CLI con IAM, consulta Configurazione delle impostazioni per. AWS CLI
- Per assistenza con AWS CDK, consulta Guida introduttiva a. AWS CDK

Informazioni aggiuntive

Costi

Quando si implementa questa soluzione, il Account AWS titolare potrebbe sostenere costi associati all'uso dei seguenti servizi. È importante comprendere come vengono fatturati questi servizi in modo da essere a conoscenza di eventuali addebiti. Per informazioni sui prezzi, consulta le pagine seguenti:

- Prezzi di Amazon SES
- Prezzi di Amazon S3
- AWS KMS prezzi
- AWS Lambda prezzi
- Prezzi di Amazon DynamoDB

Configura la risoluzione DNS per reti ibride in un ambiente AWS con account singolo

Abdullahi Olaoye, Amazon Web Services

Riepilogo

Questo modello descrive come configurare un'architettura DNS (Domain Name System) completamente ibrida che consenta la risoluzione end-to-end DNS di risorse locali, risorse AWS e query DNS Internet, senza sovraccarico amministrativo. Il modello descrive come configurare le regole di inoltro di Amazon Route 53 Resolver che determinano dove inviare una query DNS proveniente da AWS, in base al nome di dominio. Le query DNS per le risorse locali vengono inoltrate ai resolver DNS locali. Le query DNS per le risorse AWS e le query DNS Internet vengono risolte da Route 53 Resolver.

Questo modello copre la risoluzione DNS ibrida in un ambiente AWS con account singolo. Per informazioni sulla configurazione di query DNS in uscita in un ambiente AWS multi-account, consulta lo schema Configurare la risoluzione DNS per reti ibride in un ambiente AWS multi-account.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS
- Un cloud privato virtuale (VPC) nel tuo account AWS
- Una connessione di rete tra l'ambiente locale e il tuo VPC, tramite AWS Virtual Private Network (AWS VPN) o AWS Direct Connect
- Indirizzi IP dei tuoi resolver DNS locali (raggiungibili dal tuo VPC)
- Nome di dominio/sottodominio da inoltrare ai resolver locali (ad esempio, onprem.mydc.com)
- Nome di dominio/sottodominio per la zona ospitata privata AWS (ad esempio, myvpc.cloud.com)

Architecture

Stack tecnologico Target

- Zona ospitata privata Amazon Route 53
- Amazon Route 53 Resolver

- Amazon VPC
- AWS VPN o Direct Connect

Architettura Target

Tools (Strumenti)

- Amazon Route 53 Resolver semplifica il cloud ibrido per i clienti aziendali abilitando una risoluzione delle query DNS senza interruzioni sull'intero cloud ibrido. Puoi creare endpoint DNS e regole di inoltro condizionale per risolvere i namespace DNS tra il tuo data center locale e il tuo. VPCs
- La zona ospitata privata di Amazon Route 53 è un contenitore che contiene informazioni su come desideri che Route 53 risponda alle query DNS per un dominio e i relativi sottodomini all'interno di uno o più VPCs domini creati con il servizio Amazon VPC.

Epiche

Configura una zona ospitata privata

Operazione	Description	Competenze richieste
Crea una zona ospitata privata Route 53 per un nome di dominio riservato AWS come myvpc.cloud.com.	Questa zona contiene i record DNS per le risorse AWS che devono essere risolti dall'ambi ente locale. Per istruzioni, consulta Creazione di una zona ospitata privata nella documentazione di Route 53.	Amministratore di rete, amministratore di sistema
Associa la zona ospitata privata al tuo VPC.	Per consentire alle risorse del tuo VPC di risolvere i record DNS in questa zona ospitata privata, devi associare il tuo VPC alla zona ospitata. Per istruzioni, consulta Creazione di una zona ospitata privata	Amministratore di rete, amministratore di sistema

Operazione	Description	Competenze richieste
	nella documentazione di Route 53.	

Configurazione degli endpoint Route 53 Resolver

Operazione	Description	Competenze richieste
Crea un endpoint in entrata.	Route 53 Resolver utilizza l'endpoint in entrata per ricevere le query DNS dai resolver DNS locali. Per istruzioni, consulta Inoltro delle query DNS in entrata al tuo indirizzo nella documentazione di VPCs Route 53. Prendi nota dell'indirizzo IP dell'endpoint in entrata.	Amministratore di rete, amministratore di sistema
Crea un endpoint in uscita.	Route 53 Resolver utilizza l'endpoint in uscita per inviare query DNS ai resolver DNS locali. Per istruzioni, consulta Inoltro delle query DNS in uscita alla rete nella documentazione di Route 53. Prendi nota dell'ID dell'endpoint di output.	Amministratore di rete, amministratore di sistema

Imposta una regola di inoltro e associala al tuo VPC

Operazione	Description	Competenze richieste
Crea una regola di inoltro per il dominio locale.	Questa regola indicherà a Route 53 Resolver di	Amministratore di rete, amministratore di sistema

Operazione	Description	Competenze richieste
	inoltrare qualsiasi query DNS per i domini locali (come onprem.mydc.com) ai resolver DNS locali. Per creare questa regola, sono necessari gli indirizzi IP dei resolver DNS locali e l'ID dell'endpoint in uscita per Route 53 Resolver. Per istruzioni, consulta Gestione delle regole di inoltro nella documentazione di Route 53.	
Associa la regola di inoltro al tuo VPC.	Affinché la regola di inoltro abbia effetto, devi associarla al tuo VPC. Route 53 Resolver prende quindi in considera zione la regola durante la risoluzione di un dominio. Per istruzioni, consulta Gestione delle regole di inoltro nella documentazione di Route 53.	Amministratore di rete, amministratore di sistema

Configura i resolver DNS locali

Operazione	Description	Competenze richieste
Configura l'inoltro condizionale nei resolver DNS locali.	Per inviare le query DNS alla zona ospitata privata Route 53 dall'ambiente locale, è necessario configurare l'inoltro condizionale nei resolver DNS locali. Ciò indica ai resolver DNS di inoltrare tutte le query DNS per il	Amministratore di rete, amministratore di sistema

Operazione	Description	Competenze richieste
	dominio AWS (ad esempio, per myvpc.cloud.com) all'indirizzo IP dell'endpoint in entrata per Route 53 Resolver.	

Verifica la end-to-end risoluzione DNS

Operazione	Description	Competenze richieste
Testa la risoluzione DNS da AWS all'ambiente locale.	Da un server nel VPC, esegui una query DNS per un dominio locale (ad esempio server1.onprem.mydc.com).	Amministratore di rete, amministratore di sistema
Testa la risoluzione DNS dall'ambiente locale ad AWS.	Da un server locale, esegui la risoluzione DNS per un dominio AWS (ad esempio server1.myvpc.cloud.com).	Amministratore di rete, amministratore di sistema

Risorse correlate

- Gestione DNS centralizzata del cloud ibrido con Amazon Route 53 e AWS Transit Gateway (blog AWS Networking & Content Delivery)
- Semplifica la gestione DNS in un ambiente multi-account con Route 53 Resolver (blog di AWS Security)
- Utilizzo di zone ospitate private (documentazione Route 53)
- Guida introduttiva a Route 53 Resolver (documentazione Route 53)

Configura automaticamente i bot UiPath RPA su Amazon EC2 utilizzando AWS CloudFormation

Dott. Rahul Sharad Gaikwad e Tamilselvan P, Amazon Web Services

Riepilogo

Questo modello spiega come distribuire bot di automazione dei processi robotici (RPA) su istanze Amazon Elastic Compute Cloud (EC2Amazon). Utilizza una pipeline EC2 Image Builder per creare un'Amazon Machine Image (AMI) personalizzata. Un'AMI è un'immagine di macchina virtuale (VM) preconfigurata che contiene il sistema operativo (OS) e il software preinstallato per distribuire le istanze. EC2 Questo modello utilizza CloudFormation modelli AWS per installare l'edizione UiPath Studio Community sull'AMI personalizzata. UiPath è uno strumento RPA che ti aiuta a configurare robot per automatizzare le tue attività.

Come parte di questa soluzione, le istanze di EC2 Windows vengono avviate utilizzando l'AMI di base e l'applicazione UiPath Studio viene installata sulle istanze. Il modello utilizza lo strumento Microsoft System Preparation (Sysprep) per duplicare l'installazione personalizzata di Windows. Dopodiché, rimuove le informazioni sull'host e crea un AMI finale dall'istanza. È quindi possibile avviare le istanze su richiesta utilizzando l'AMI finale con le proprie convenzioni di denominazione e configurazione di monitoraggio.

> Nota: questo modello non fornisce alcuna informazione sull'utilizzo dei bot RPA. Per queste informazioni, consulta la UiPath documentazione. È inoltre possibile utilizzare questo modello per configurare altre applicazioni bot RPA personalizzando i passaggi di installazione in base alle proprie esigenze.

Questo modello offre le seguenti automazioni e vantaggi:

- Distribuzione e condivisione di applicazioni: puoi creare Amazon EC2 AMIs per la distribuzione di applicazioni e condividerle su più account tramite una pipeline EC2 Image Builder, che utilizza i CloudFormation modelli AWS come script Infrastructure as Code (IaC).
- EC2 Provisioning e scalabilità di Amazon: i modelli CloudFormation IaC forniscono sequenze di nomi di computer personalizzate e l'automazione dei join di Active Directory.
- Osservabilità e monitoraggio: il modello configura i CloudWatch dashboard di Amazon per aiutarti a monitorare i EC2 parametri di Amazon (come l'utilizzo della CPU e del disco).

- Vantaggi dell'RPA per la tua azienda: l'RPA migliora la precisione perché i robot possono eseguire le attività assegnate in modo automatico e coerente. La RPA aumenta anche la velocità e la produttività perché elimina le operazioni che non aggiungono valore e gestisce attività ripetitive.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Autorizzazioni AWS Identity and Access Management (IAM) per la distribuzione CloudFormation di modelli
- Politiche IAM per configurare la distribuzione AMI tra account con EC2 Image Builder

Architecture

1. L'amministratore fornisce l'AMI Windows di base nel `ec2-image-builder.yaml` file e distribuisce lo stack nella CloudFormation console.
2. Lo CloudFormation stack implementa la EC2 pipeline Image Builder, che include le seguenti risorse:
   - `Ec2ImageInfraConfiguration`
   - `Ec2ImageComponent`
   - `Ec2ImageRecipe`
   - `Ec2AMI`
3. La pipeline EC2 Image Builder avvia un' EC2 istanza temporanea di Windows utilizzando l'AMI di base e installa i componenti richiesti (in questo caso, Studio). UiPath
4. EC2 Image Builder rimuove tutte le informazioni sull'host e crea un'AMI da Windows Server.
5. Aggiorna il `ec2-provisioning yaml` file con l'AMI personalizzata e avvii una serie di EC2 istanze in base alle tue esigenze.
6. La macro Count viene distribuita utilizzando un CloudFormation modello. Questa macro fornisce una proprietà Count per CloudFormation le risorse che consente di specificare facilmente più risorse dello stesso tipo.
7. Si aggiorna il nome della macro nel CloudFormation `ec2-provisioning.yaml` file e si distribuisce lo stack.

8. L'amministratore aggiorna il `ec2-provisioning.yaml` file in base ai requisiti e avvia lo stack.
9. Il modello distribuisce le EC2 istanze con l'applicazione Studio. UiPath

Tools (Strumenti)

Servizi AWS

- AWS ti CloudFormation aiuta a modellare e gestire le risorse dell'infrastruttura in modo automatizzato e sicuro.
- Amazon ti CloudWatch aiuta a osservare e monitorare risorse e applicazioni su AWS, on-premise e su altri cloud.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo sicura e ridimensionabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- EC2 Image Builder semplifica la creazione, il test e la distribuzione di macchine virtuali e immagini di container da utilizzare su AWS o in locale.
- Amazon ti EventBridge aiuta a creare applicazioni basate sugli eventi su larga scala su AWS, sistemi esistenti o applicazioni Software as a Service (SaaS).
- AWS Identity and Access Management (IAM) ti aiuta a controllare in modo sicuro l'accesso alle risorse AWS. Con IAM, puoi gestire centralmente le autorizzazioni che controllano a quali risorse AWS possono accedere gli utenti. Utilizza IAM per controllare chi è autenticato (accesso effettuato) e autorizzato (dispone di autorizzazioni) per l'utilizzo di risorse.
- AWS Lambda è un servizio di elaborazione serverless e basato sugli eventi che consente di eseguire codice per praticamente qualsiasi tipo di applicazione o servizio di backend senza dover fornire o gestire server. Puoi richiamare le funzioni Lambda da oltre 200 servizi AWS e applicazioni SaaS e pagare solo per ciò che usi.
- Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è un servizio di storage di oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Systems Manager Agent (SSM Agent) aiuta Systems Manager ad aggiornare, gestire e configurare EC2 istanze, dispositivi edge, server locali e macchine virtuali (). VMs

Archivi di codice

Il codice per questo pattern è disponibile nella configurazione del bot GitHub UiPath RPA utilizzando CloudFormation il repository. Il modello utilizza anche una macro disponibile nel repository AWS CloudFormation Macros.

Best practice

- AWS rilascia nuovi Windows AMIs ogni mese. Questi contengono le patch, i driver e gli agenti di lancio più recenti del sistema operativo. Ti consigliamo di utilizzare l'AMI più recente quando avvii nuove istanze o quando crei immagini personalizzate.
- Applica tutte le patch di sicurezza Windows o Linux disponibili durante la creazione delle immagini.

Epiche

Implementa una pipeline di immagini per l'immagine di base

Operazione	Description	Competenze richieste
Configura una pipeline di EC2 Image Builder.	1. Clona la configurazione del bot UiPath RPA utilizzando il CloudFormation repository o scarica il `ec2-image-builder.yaml` modello dal repository. 2. Accedi alla Console di gestione AWS e apri la CloudFormation console AWS. 3. Seleziona Crea stack. 4. Nella sezione Specify template(Specifica il modello) scegliere Upload a template file (Carica un file modello). 5. Individua e carica il `ec2-image-builder.yaml`	AWS DevOps

Operazione	Description	Competenze richieste
	modello dal tuo computer, quindi scegli Avanti. 6. Fornisci i parametri di input per il tuo stack o accetta i valori predefiniti. Scegli Next (Successivo). **Note** Il numero e i valori dei parametri possono variare a seconda dei valori di input. 7. Facoltativamente, configura le opzioni dello stack, quindi scegli Avanti. 8. Controlla i dettagli dello stack. 9. Alla fine della schermata , seleziona la casella di controllo per confermare le funzionalità, quindi scegli Invia. 10. Monitora l'avanzamento dello stack. Quando lo stato è`CREATE_COMPLETE` , la distribuzione è pronta.	

Operazione	Description	Competenze richieste
Visualizza le impostazioni di EC2 Image Builder.	Le impostazioni di EC2 Image Builder includono la configurazione dell'infrastruttura, le impostazioni di distribuzione e le impostazioni di scansione di sicurezza. Per visualizzare le impostazioni: 1. Aprire la console EC2 Image Builder. 2. Dal pannello di navigazione, accedete alle varie impostazioni di Image Builder. **Note** Come procedura consigliata, è consigliabile apportare eventuali aggiornamenti a EC2 Image Builder solo tramite il CloudFormation modello.	AWS DevOps

Operazione	Description	Competenze richieste
Visualizza la pipeline di immagini.	Per visualizzare la pipeline di immagini distribuita: 1. Nella console EC2 Image Builder, scegliete Image pipelines dal pannello di navigazione. 2. Seleziona la pipeline di immagini che hai creato. 3. Visualizza i dettagli di configurazione delle immagini di output, la ricetta dell'immagine, la configura zione dell'infrastruttura, le impostazioni di distribuz ione, EventBridge le regole di Amazon e i tag.	AWS DevOps

Operazione	Description	Competenze richieste
Visualizza i log di Image Builder.	EC2 I log di Image Builder sono aggregati in gruppi di log. CloudWatch Per visualizzare i log in: CloudWatch 1. Apri la CloudWatch console. 2. Nel pannello di navigazione a sinistra, scegli Log, Gruppi di log. 3. Scegli il nome del gruppo di log. EC2 I log di Image Builder vengono aggregati nel gruppo di log. `/aws/imagebuilder/XXX` 4. Controllate i log più recenti nel rispettivo flusso di log per eventuali errori riscontrati durante l'esecuzione della pipeline di immagini. EC2 I log di Image Builder vengono inoltre archiviati in un bucket S3. Per visualizzare i log nel bucket: 1. Apri la console Amazon S3. 2. Nell'elenco Bucket, seleziona il nome del bucket. I log vengono aggregati nel bucket S3. `<stack-name>-XXXXX X`	AWS DevOps

Operazione	Description	Competenze richieste
Carica il UiPath file in un bucket S3.	1. Scarica il `.msi` file per UiPath Studio dalla posizione https://download.uipath.com UiPathStudioCommunity/.msi. 2. Caricare il file in un bucket S3. 3. Aggiorna il nome del bucket e la chiave del file nel `ec2-image-builder.yaml` modello, nella sezione dati utente, riga numero 310.	AWS DevOps

Implementa e testa la macro Count

Operazione	Description	Competenze richieste
Implementa la macro Count.	1. Clona o scarica la macro Count CloudFormation . 2. Accedi alla cartella `Count`. 3. Avrai bisogno di un bucket S3 per archiviare gli artefatti . CloudFormation Se non disponi già di un bucket S3, creane uno con il nome. `aws s3 mb s3://<bucket name>` 4. Package del modello di macro Count. Il modello utilizza AWS Serverless Application Model (SAM), quindi deve essere	DevOps ingegnere

Operazione	Description	Competenze richieste

	trasformato prima di poterlo distribuire.
	
	`aws cloudformation package \ --template-file template.yaml \ --s3-bucket <your bucket name here> \ --output-template-file packaged.yaml`
	
	Esempio:
	
	`aws cloudformation package \ --template-file template.yaml \ --s3-bucket count-macro-ec2 \ --output-template-file packaged.yaml`
	
	5. Implementa il modello confezionato per creare uno stack. CloudFormation
	
	`aws cloudformation deploy \ --stack-name Count-macro \ --template-file packaged.yaml \ --capabilities CAPABILITY_IAM`	

Operazione	Description	Competenze richieste
	Se vuoi usare la console, segui le istruzioni nell'epic precedente o nella documentazione. CloudFormation	
Prova la macro Count.	Per testare le funzionalità della macro, prova ad avviare il modello di esempio fornito con la macro. `aws cloudformation deploy \ --stack-name Count-test \ --template-file test.yaml \ --capabilities CAPABILITY_IAM`	DevOps ingegnere

Implementa lo CloudFormation stack per fornire alle istanze l'immagine personalizzata

Operazione	Description	Competenze richieste
Implementa il modello di EC2 provisioning di Amazon.	Per distribuire EC2 Image Pipeline utilizzando: CloudFormation 1. Scarica il `ec2-provisioning.yaml` modello dal GitHub repository o localizzalo sul tuo computer se hai clonato il repository. 2. Apri la CloudFormation console.	AWS DevOps

Operazione	Description	Competenze richieste
	3. Ripeti i passaggi del primo epic (o segui le istruzioni nella documentazione) per distribuirlo. CloudFormation `ec2-provisioning.yaml`	
Visualizza EC2 le impostazioni di Amazon.	Le EC2 impostazioni di Amazon includono sicurezza, rete, archiviazione, controlli dello stato, monitoraggio e configurazioni di tag. Per visualizzare queste configurazioni: 1. Apri la EC2 console Amazon. 2. Nel riquadro di navigazione, scegli Istanze, quindi seleziona l' EC2 istanza creata dal modello di EC2 provisioning di Amazon. 3. Nel riepilogo dell'istanza, seleziona le schede per visualizzare le EC2 impostazioni Amazon corrispondenti.	AWS DevOps

Operazione	Description	Competenze richieste
Visualizza la CloudWatch dashboard.	1. Apri la CloudWatch console. 2. Nel pannello di navigazione seleziona Dashboards (Pannelli di controllo). 3. Scegli la dashboard con il nome del tuo stack. **Note** Dopo aver effettuato il provisioning dello stack, è necessario del tempo per compilare la dashboard con le metriche. La dashboard fornisce le seguenti metriche:`CPUUtilization` ,,,,`DiskUtilization` ,`MemoryUtilization` . `NetworkIn` `NetworkOut` `StatusCheckFailed`	AWS DevOps

Operazione	Description	Competenze richieste
Visualizza metriche personalizzate per l'utilizzo della memoria e del disco.	1. Sulla CloudWatch console, scegli Dashboard. 2. Nel pannello di navigazione, seleziona Metrics (Parametri), All metrics (Tutti i parametri). 3. Scegli Namespace personalizzati,. CWAgent	AWS DevOps
Visualizza gli allarmi relativi all'utilizzo della memoria e del disco.	1. Sulla CloudWatch console, nel riquadro di navigazione, scegli Dashboard. 2. Scegli All alarms (Tutti gli allarmi).	AWS DevOps
Verifica la regola del ciclo di vita delle istantanee.	1. Apri la EC2 console Amazon. 2. Nel pannello di navigazione, seleziona Lifecycle Manager. 3. Verifica le impostazioni per il ciclo di vita dell'AMI.	AWS DevOps

Eliminare l'ambiente (opzionale)

Operazione	Description	Competenze richieste
Eliminare le pile.	Una volta completato il PoC o il progetto pilota, ti consigliamo di eliminare gli stack che hai creato per assicurarti che non ti vengano addebitati costi per queste risorse.	AWS DevOps

Operazione	Description	Competenze richieste
	1. Apri la CloudFormation console AWS. 2. Nel pannello di navigazione, scegli Stacks, quindi seleziona uno o entrambi gli stack creati in precedenza che desideri eliminare. Lo stack deve essere attualmente in esecuzione. 3. Nel riquadro dei dettagli dello stack, scegliere Delete (Elimina). 4. Quando viene richiesto, scegliere nuovamente Delete stack (Elimina stack). **Important** L'operazione di eliminazione dello stack non può essere interrotta dopo l'inizio. Lo stack procede allo stato `DELETE_IN_PROGRESS`. Se l'eliminazione fallisce, lo stack si troverà nello stato in cui si `DELETE_FAILED` trova. Per le soluzioni, consulta Delete stack fail nella documentazione CloudForm	

Operazione	Description	Competenze richieste
	ation sulla risoluzione dei problemi di AWS. Per informazioni sulla protezione degli stack dall'elim inazione accidentale, consulta Proteggere uno stack dall'elim inazione nella documenta zione AWS. CloudFormation	

risoluzione dei problemi

Problema	Soluzione
Quando distribuisci il modello di EC2 provision ing Amazon, ricevi l'errore: Risposta non valida ricevuta da transform 123xxxx: :Count.	Si tratta di un problema noto. (Vedi la soluzione personalizzata e PR nel repository di CloudFormation macro AWS.) Per risolvere questo problema, apri la console AWS Lambda e aggiorna `index.py` con il contenuto del repository. GitHub

Risorse correlate

GitHub repository

- UiPath configurazione del bot RPA utilizzando CloudFormation
- Conta Macro CloudFormation

Riferimenti AWS

- Creazione di uno stack sulla CloudFormation console AWS (CloudFormation documentazione)
- Risoluzione dei problemi CloudFormation (CloudFormation documentazione)

- Monitora i parametri di memoria e disco per le EC2 istanze Amazon (documentazione Amazon) EC2
- Come posso usare l' CloudWatch agente per visualizzare i parametri di Performance Monitor su un server Windows? (Articolo AWS Re:Post)

Riferimenti aggiuntivi

- UiPath documentazione
- Impostazione del nome host in un SysPreped AMI (post sul blog di Brian Beach)
- Come faccio a fare in modo che Cloudformation rielabori un modello utilizzando una macro quando i parametri cambiano? (Stack Overflow)

Configura un' PeopleSoft architettura ad alta disponibilità su AWS

Ramanathan Muralidhar, Amazon Web Services

Riepilogo

Quando esegui la migrazione dei PeopleSoft carichi di lavoro in AWS, la resilienza è un obiettivo importante. Garantisce che la tua PeopleSoft applicazione sia sempre altamente disponibile e in grado di ripristinarsi rapidamente in caso di guasti.

Questo modello fornisce un'architettura per PeopleSoft le tue applicazioni su AWS per garantire l'alta disponibilità (HA) a livello di rete, applicazione e database. Utilizza un database Amazon Relational Database Service (Amazon RDS) per Oracle o Amazon RDS for SQL Server per il livello del database. Questa architettura include anche servizi AWS come Amazon Route 53, Amazon Elastic Compute Cloud (Amazon EC2), istanze Linux, Amazon Elastic Block Storage (Amazon EBS), Amazon Elastic File System (Amazon EFS) e Application Load Balancer ed è scalabile.

Oracle PeopleSoft offre una suite di strumenti e applicazioni per la gestione della forza lavoro e altre operazioni aziendali.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un PeopleSoft ambiente con le licenze necessarie per configurarlo su AWS
- Un cloud privato virtuale (VPC) configurato nel tuo account AWS con le seguenti risorse:
  - Almeno due zone di disponibilità
  - Una sottorete pubblica e tre sottoreti private in ogni zona di disponibilità
  - Un gateway NAT e un gateway Internet
  - Tabelle di routing per ogni sottorete per instradare il traffico
  - Elenchi di controllo dell'accesso alla rete (rete ACLs) e gruppi di sicurezza definiti per garantire la sicurezza dell' PeopleSoft applicazione in conformità con gli standard dell'organizzazione

Limitazioni

- Questo modello fornisce una soluzione ad alta disponibilità (HA). Non supporta scenari di disaster recovery (DR). Nel raro caso in cui l'intera regione AWS per l'implementazione HA non funzioni, l'applicazione non sarà più disponibile.

Versioni del prodotto

- PeopleSoft applicazioni che eseguono PeopleTools 8.52 e versioni successive

Architecture

Architettura Target

I tempi di inattività o le interruzioni dell'applicazione di PeopleSoft produzione influiscono sulla disponibilità dell'applicazione e causano gravi interruzioni dell'attività.

Si consiglia di progettare l'applicazione PeopleSoft di produzione in modo che sia sempre altamente disponibile. È possibile raggiungere questo obiettivo eliminando i singoli punti di errore, aggiungendo punti di crossover o failover affidabili e rilevando i guasti. Il diagramma seguente illustra un'architettura HA per on PeopleSoft AWS.

Questa implementazione di architettura utilizza Amazon RDS for PeopleSoft Oracle come database e istanze EC2 in esecuzione su Red Hat Enterprise Linux (RHEL). Puoi anche usare Amazon RDS for SQL Server come database Peoplesoft.

Questa architettura contiene i seguenti componenti:

- Amazon Route 53 viene utilizzato come Domain Name Server (DNS) per il routing delle richieste da Internet all' PeopleSoft applicazione.
- AWS WAF ti aiuta a proteggerti da exploit e bot Web comuni che possono influire sulla disponibilità, compromettere la sicurezza o consumare risorse eccessive. AWS Shield Advanced (non illustrato) offre una protezione molto più ampia.
- Un Application Load Balancer bilancia il carico del traffico HTTP e HTTPS con un routing avanzato delle richieste rivolto ai server Web.
- I server Web, i server delle applicazioni, i server di pianificazione dei processi e i server Elasticsearch che supportano l' PeopleSoft applicazione vengono eseguiti in più zone di disponibilità e utilizzano Amazon EC2 Auto Scaling.

- Il database utilizzato dall' PeopleSoft applicazione viene eseguito su Amazon RDS in una configurazione Multi-AZ.
- La condivisione di file utilizzata dall' PeopleSoft applicazione è configurata su Amazon EFS e viene utilizzata per accedere ai file tra le istanze.
- Amazon Machine Images (AMI) vengono utilizzate da Amazon EC2 Auto Scaling per PeopleSoft garantire che i componenti vengano clonati rapidamente quando necessario.
- I gateway NAT collegano le istanze in una sottorete privata a servizi esterni al VPC e assicurano che i servizi esterni non possano avviare una connessione con tali istanze.
- Il gateway Internet è un componente VPC scalabile orizzontalmente, ridondante e ad alta disponibilità che consente la comunicazione tra il tuo VPC e Internet.
- Gli host bastion nella sottorete pubblica forniscono l'accesso ai server nella sottorete privata da una rete esterna, come Internet o una rete locale. Gli host bastion forniscono un accesso controllato e sicuro ai server nelle sottoreti private.

Dettagli dell'architettura

Il PeopleSoft database è ospitato in un database Amazon RDS for Oracle (o Amazon RDS for SQL Server) in una configurazione Multi-AZ. La funzionalità Amazon RDS Multi-AZ replica gli aggiornamenti del database su due zone di disponibilità per aumentare la durabilità e la disponibilità. Amazon RDS esegue automaticamente il failover nel database di standby per la manutenzione pianificata e le interruzioni non pianificate.

Il livello PeopleSoft Web e quello intermedio vengono installati sulle istanze EC2. Queste istanze sono distribuite su più zone di disponibilità e collegate da un gruppo di Auto Scaling. Ciò garantisce che questi componenti siano sempre altamente disponibili. Viene mantenuto un numero minimo di istanze richieste per garantire che l'applicazione sia sempre disponibile e possa scalare quando necessario.

Si consiglia di utilizzare un tipo di istanza EC2 di ultima generazione per le istanze EC2 OEM. I tipi di istanze dell'attuale generazione, come le istanze create su AWS Nitro System, supportano macchine virtuali hardware ()HVMs. Gli HVM AMIs sono necessari per sfruttare i vantaggi del networking avanzato e offrono anche una maggiore sicurezza. Le istanze EC2 che fanno parte di ciascun gruppo Auto Scaling utilizzano la propria AMI quando sostituiscono o aumentano le istanze. Ti consigliamo di selezionare i tipi di istanza EC2 in base al carico che desideri che l' PeopleSoft applicazione gestisca e ai valori minimi consigliati da Oracle per l'applicazione e la versione. PeopleSoft PeopleTools Per ulteriori informazioni sui requisiti hardware e software, consulta il sito Web di supporto Oracle.

Il PeopleSoft web e il livello intermedio condividono un mount Amazon EFS per condividere report, file di dati e (se necessario) la `PS_HOME` directory. Amazon EFS è configurato con obiettivi di montaggio in ogni zona di disponibilità per motivi di prestazioni e costi.

Viene fornito un Application Load Balancer per supportare il traffico che accede all' PeopleSoft applicazione e bilancia il carico tra i server Web in diverse zone di disponibilità. Un Application Load Balancer è un dispositivo di rete che fornisce HA in almeno due zone di disponibilità. I server Web distribuiscono il traffico su diversi server di applicazioni utilizzando una configurazione di bilanciamento del carico. Il bilanciamento del carico tra il server Web e il server delle applicazioni assicura che il carico sia distribuito in modo uniforme tra le istanze e aiuta a evitare colli di bottiglia e interruzioni del servizio dovute al sovraccarico delle istanze.

Amazon Route 53 viene utilizzato come servizio DNS per indirizzare il traffico verso l'Application Load Balancer da Internet. Route 53 è un servizio Web DNS altamente scalabile e disponibile.

Dettagli HA

- Database: la funzionalità Multi-AZ di Amazon RDS gestisce due database in più zone di disponibilità con replica sincrona. Questo crea un ambiente ad alta disponibilità con failover automatico. Amazon RDS dispone del rilevamento degli eventi di failover e avvia il failover automatico quando si verificano questi eventi. Puoi anche avviare il failover manuale tramite l'API Amazon RDS. Per una spiegazione dettagliata, consulta il post sul blog Amazon RDS Under The Hood: Multi-AZ. Il failover è semplice e l'applicazione si riconnette automaticamente al database quando si verifica. Tuttavia, tutti i job del Process Scheduler durante il failover generano errori e devono essere inviati nuovamente.
- PeopleSoft server delle applicazioni: i server delle applicazioni sono distribuiti su più zone di disponibilità e dispongono di un gruppo Auto Scaling definito per loro. Se un'istanza fallisce, il gruppo Auto Scaling la sostituisce immediatamente con un'istanza integra clonata dall'AMI del modello di Application Server. In particolare, il jolt pooling è abilitato, quindi quando un'istanza del server delle applicazioni si interrompe, le sessioni eseguono automaticamente il failover su un altro server delle applicazioni e il gruppo Auto Scaling avvia automaticamente un'altra istanza, richiama il server delle applicazioni e lo registra nel mount Amazon EFS. L'application server appena creato viene aggiunto automaticamente ai server Web utilizzando `PSSTRSETUP.SH` lo script nei server Web. Ciò garantisce che il server delle applicazioni sia sempre altamente disponibile e si ripristini rapidamente in caso di guasto.
- Process scheduler: i server Process schedulers sono distribuiti su più zone di disponibilità e dispongono di un gruppo Auto Scaling definito per loro. Se un'istanza fallisce, il gruppo Auto Scaling la sostituisce immediatamente con un'istanza integra clonata dall'AMI del modello del

server Process Scheduler. In particolare, quando un'istanza di Process Scheduler si interrompe, il gruppo Auto Scaling attiva automaticamente un'altra istanza e attiva lo scheduler di processo. Tutti i processi in esecuzione quando l'istanza ha avuto esito negativo devono essere inoltrati nuovamente. Ciò garantisce che il Process Scheduler sia sempre altamente disponibile e si ripristini rapidamente in caso di guasto.

- Server Elasticsearch: per i server Elasticsearch è definito un gruppo Auto Scaling. Se un'istanza fallisce, il gruppo Auto Scaling la sostituisce immediatamente con un'istanza integra clonata dall'AMI del modello di server Elasticsearch. In particolare, quando un'istanza Elasticsearch si interrompe, l'Application Load Balancer che invia le richieste rileva l'errore e interrompe l'invio di traffico verso di essa. Il gruppo Auto Scaling avvia automaticamente un'altra istanza e fa apparire l'istanza Elasticsearch. Quando l'istanza Elasticsearch viene ripristinata, l'Application Load Balancer rileva che è integra e ricomincia a inviarle richieste. Ciò garantisce che il server Elasticsearch sia sempre altamente disponibile e si ripristini rapidamente in caso di guasto.
- Server Web: per i server Web è definito un gruppo Auto Scaling. Se un'istanza fallisce, il gruppo Auto Scaling la sostituisce immediatamente con un'istanza integra clonata dall'AMI del modello del server Web. In particolare, quando un'istanza del server Web si interrompe, l'Application Load Balancer che invia le richieste rileva l'errore e interrompe l'invio di traffico verso di essa. Il gruppo Auto Scaling avvia automaticamente un'altra istanza e visualizza l'istanza del server Web. Quando viene eseguito il backup dell'istanza del server Web, l'Application Load Balancer rileva che è integra e ricomincia a inviarle richieste. Ciò garantisce che il server Web sia sempre altamente disponibile e si ripristini rapidamente in caso di guasto.

Tools (Strumenti)

Servizi AWS

- Gli Application Load Balancer distribuiscono il traffico delle applicazioni in entrata su più destinazioni, come le istanze EC2, in più zone di disponibilità.
- Amazon Elastic Block Store (Amazon EBS) fornisce volumi di storage a livello di blocco da utilizzare con le istanze Amazon Elastic Compute Cloud (Amazon EC2).
- Amazon Elastic Compute Cloud (Amazon EC2) Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon Elastic File System (Amazon EFS) ti aiuta a creare e configurare file system condivisi nel cloud AWS.

- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS.
- Amazon Route 53 è un servizio Web DNS altamente scalabile e disponibile.

Best practice

Best practice operative

- Quando esegui PeopleSoft su AWS, usa Route 53 per indirizzare il traffico da Internet e localmente. Utilizza l'opzione di failover per reindirizzare il traffico verso il sito di disaster recovery (DR) se l'istanza DB principale non è disponibile.
- Usa sempre un Application Load Balancer davanti all' PeopleSoft ambiente. Ciò garantisce che il traffico venga bilanciato in modo sicuro in termini di carico verso i server Web.
- Nelle impostazioni del gruppo target di Application Load Balancer, assicurati che la viscosità sia attivata con un cookie generato dal load balancer.

> **Note**
>
> Potrebbe essere necessario utilizzare un cookie basato sull'applicazione se utilizzi il Single Sign-On (SSO) esterno. Ciò garantisce che le connessioni siano coerenti tra i server Web e i server delle applicazioni.

- Per un'applicazione PeopleSoft di produzione, il timeout di inattività di Application Load Balancer deve corrispondere a quello impostato nel profilo Web utilizzato. Ciò impedisce la scadenza delle sessioni utente a livello di bilanciamento del carico.
- Per un'applicazione PeopleSoft di produzione, impostate il numero di riciclo del server delle applicazioni su un valore che riduca al minimo le perdite di memoria.
- Se utilizzi un database Amazon RDS per la tua applicazione di PeopleSoft produzione, come descritto in questo modello, eseguilo in formato Multi-AZ per un'elevata disponibilità.
- Se il database è in esecuzione su un'istanza EC2 per l'applicazione di PeopleSoft produzione, assicurati che un database in standby sia in esecuzione su un'altra zona di disponibilità per un'elevata disponibilità.
- Per il DR, assicurati che il tuo database Amazon RDS o l'istanza EC2 abbiano uno standby configurato in una regione AWS separata dal database di produzione. Ciò garantisce che, in caso di emergenza nella regione, sia possibile trasferire l'applicazione in un'altra regione.

- Per il DR, usa Amazon Elastic Disaster Recovery per configurare componenti a livello di applicazione in una regione separata dai componenti di produzione. Ciò garantisce che, in caso di emergenza nella regione, sia possibile trasferire l'applicazione in un'altra regione.
- Usa Amazon EFS (per I/O requisiti moderati) o Amazon FSx (per I/O requisiti elevati) per archiviare PeopleSoft report, allegati e file di dati. Ciò garantisce che i contenuti siano archiviati in un'unica posizione centrale e siano accessibili da qualsiasi punto all'interno dell'infrastruttura.
- Usa Amazon CloudWatch (di base e dettagliato) per monitorare quasi in tempo reale le risorse del cloud AWS utilizzate dalla tua PeopleSoft applicazione. In questo modo sarai avvisato istantaneamente dei problemi e potrai risolverli rapidamente prima che influiscano sulla disponibilità dell'ambiente.
- Se utilizzi un database Amazon RDS come PeopleSoft database, utilizza Enhanced Monitoring. Questa funzionalità fornisce l'accesso a oltre 50 parametri, tra cui CPU, memoria e file system. I/O, and disk I/O
- Usa AWS CloudTrail per monitorare le chiamate API sulle risorse AWS utilizzate dalla tua PeopleSoft applicazione. Questo ti aiuta a eseguire analisi di sicurezza, tracciare le modifiche alle risorse e verificare la conformità.

Le migliori pratiche di sicurezza

- Per proteggere la tua PeopleSoft applicazione da exploit comuni come SQL injection o cross-site scripting (XSS), usa AWS WAF. Prendi in considerazione l'utilizzo di AWS Shield Advanced per servizi di rilevamento e mitigazione personalizzati.
- Aggiungi una regola all'Application Load Balancer per reindirizzare automaticamente il traffico da HTTP a HTTPS per proteggere la tua applicazione. PeopleSoft
- Configura un gruppo di sicurezza separato per Application Load Balancer. Questo gruppo di sicurezza dovrebbe consentire solo il traffico HTTPS/HTTP in entrata e nessun traffico in uscita. Ciò garantisce che sia consentito solo il traffico previsto e contribuisce a proteggere l'applicazione.
- Utilizzate sottoreti private per i server delle applicazioni, i server Web e il database e utilizzate i gateway NAT per il traffico Internet in uscita. Ciò garantisce che i server che supportano l'applicazione non siano raggiungibili pubblicamente, fornendo al contempo l'accesso pubblico solo ai server che ne hanno bisogno.
- Usalo in VPCs modo diverso per gestire i tuoi PeopleSoft ambienti di produzione e non di produzione. Usa AWS Transit Gateway, il peering VPC, la rete ACLs e i gruppi di sicurezza per controllare il flusso di traffico tra i VPC e, se necessario, il data center locale.

- Segui il principio del privilegio minimo. Concedi l'accesso alle risorse AWS utilizzate dall' PeopleSoft applicazione solo agli utenti che ne hanno assolutamente bisogno. Concedi solo i privilegi minimi necessari per eseguire un'attività. Per ulteriori informazioni, consulta il pilastro della sicurezza di AWS Well-Architected Framework.
- Ove possibile, utilizza AWS Systems Manager per accedere alle istanze EC2 utilizzate dall' PeopleSoft applicazione.

Le migliori pratiche di affidabilità

- Quando utilizzi un Application Load Balancer, registra una singola destinazione per ogni zona di disponibilità abilitata. Questo rende il load balancer più efficace.
- Ti consigliamo di averne tre distinti URLs per ogni ambiente PeopleSoft di produzione: un URL per accedere all'applicazione, uno per servire il broker di integrazione e uno per visualizzare i report. Se possibile, ogni URL dovrebbe avere i propri server Web e server applicativi dedicati. Questo design aiuta a rendere PeopleSoft l'applicazione più sicura, poiché ogni URL ha una funzionalità distinta e un accesso controllato. Inoltre, riduce al minimo l'ambito di impatto in caso di guasto dei servizi sottostanti.
- Ti consigliamo di configurare i controlli di integrità sui gruppi target del sistema di bilanciamento del carico per la tua PeopleSoft applicazione. I controlli di integrità devono essere eseguiti sui server Web anziché sulle istanze EC2 che eseguono tali server. Ciò garantisce che se il server Web si blocca o l'istanza EC2 che ospita il server Web si interrompe, l'Application Load Balancer rifletta tali informazioni in modo accurato.
- Per un'applicazione PeopleSoft di produzione, consigliamo di distribuire i server Web su almeno tre zone di disponibilità. Ciò garantisce che l' PeopleSoft applicazione sia sempre altamente disponibile anche in caso di interruzione di una delle zone di disponibilità.
- Per un'applicazione PeopleSoft di produzione, abilita jolt pooling (). `joltPooling=true` Ciò garantisce che l'applicazione esegua il failover su un altro server delle applicazioni se un server è inattivo per l'applicazione di patch o a causa di un errore di una macchina virtuale.
- Per un'applicazione PeopleSoft di produzione, impostate su `1DynamicConfigReload` . Questa impostazione è supportata nella PeopleTools versione 8.52 e successive. Aggiunge nuovi server di applicazioni al server Web in modo dinamico, senza riavviare i server.
- Per ridurre al minimo i tempi di inattività durante l'applicazione delle PeopleTools patch, utilizzate il metodo di blue/green distribuzione per le configurazioni di avvio del gruppo Auto Scaling per i server Web e applicativi. Per ulteriori informazioni, consulta il white paper Panoramica delle opzioni di distribuzione su AWS.

- Usa AWS Backup per eseguire il backup della tua PeopleSoft applicazione su AWS. Backup AWS è un servizio economico, completamente gestito e basato su policy che semplifica la protezione dei dati su larga scala.

Le migliori pratiche in termini di prestazioni

- Interrompi l'SSL presso l'Application Load Balancer per prestazioni ottimali dell'ambiente, a meno che PeopleSoft la tua azienda non richieda traffico crittografato in tutto l'ambiente.
- Crea endpoint VPC di interfaccia per servizi AWS come Amazon Simple Notification Service (Amazon SNS) in modo che il traffico CloudWatchsia sempre interno. È conveniente e aiuta a proteggere la tua applicazione.

Best practice per l'ottimizzazione dei costi

- Etichetta tutte le risorse utilizzate dal tuo PeopleSoft ambiente e abilita i tag di allocazione dei costi. Questi tag consentono di visualizzare e gestire i costi delle risorse.
- Per un'applicazione PeopleSoft di produzione, configurate i gruppi di Auto Scaling per i server Web e i server delle applicazioni. Ciò mantiene un numero minimo di server Web e applicativi per supportare l'applicazione. È possibile utilizzare le politiche di gruppo Auto Scaling per aumentare e ridurre i server in base alle esigenze.
- Utilizza gli allarmi di fatturazione per ricevere avvisi quando i costi superano una soglia di budget specificata.

Le migliori pratiche di sostenibilità

- Usa l'infrastruttura come codice (IaC) per gestire i tuoi PeopleSoft ambienti. Questo ti aiuta a creare ambienti coerenti e a mantenere il controllo delle modifiche.

Epiche

Esegui la migrazione del PeopleSoft database su Amazon RDS

Operazione	Description	Competenze richieste
Creazione di un gruppo di sottoreti del database.	Sulla console Amazon RDS, nel riquadro di navigazio	Amministratore cloud

Operazione	Description	Competenze richieste
	ne, scegli Gruppi di sottoreti , quindi crea un gruppo di sottoreti Amazon RDS DB con sottoreti in più zone di disponibilità. Ciò è necessario per l'esecuzione del database Amazon RDS in una configura zione Multi-AZ.	
Crea il database Amazon RDS.	Crea un database Amazon RDS in una zona di disponibi lità della regione AWS selezionata per l'ambient e PeopleSoft HA. Quando crei il database Amazon RDS, assicurati di seleziona re l'opzione Multi-AZ (Crea un'istanza di standby) e il gruppo di sottoreti del database creato nel passaggio precedente. Per ulteriori informazioni, consulta la documentazione di Amazon RDS.	Amministratore del cloud, amministratore del database Oracle

Operazione	Description	Competenze richieste
Esegui la migrazione del PeopleSoft database su Amazon RDS.	Esegui la migrazione del PeopleSoft database esistente nel database Amazon RDS utilizzando AWS Database Migration Service (AWS DMS). Per ulteriori informazioni, consulta la documentazione di AWS DMS e il post sul blog di AWS Migrazione dei database Oracle con tempi di inattività quasi nulli utilizzando AWS DMS.	PeopleSoft Amministratore del cloud, DBA

Configura il tuo file system Amazon EFS

Operazione	Description	Competenze richieste
Creare un file system.	Sulla console Amazon EFS, crea un file system e monta gli obiettivi per ogni zona di disponibilità. Per istruzioni, consulta la documentazione di Amazon EFS. Una volta creato il file system, annota il suo nome DNS. Queste informazioni verranno utilizzate durante il montaggio del file system.	Amministratore del cloud

Configura l' PeopleSoft applicazione e il file system

Operazione	Description	Competenze richieste
Avvio di un'istanza EC2.	Avvia un'istanza EC2 per la tua PeopleSoft applicazione. Per istruzioni, consulta la documentazione di Amazon EC2. • In Nome, inserisci `APP_TEMPLATE` . • Per le immagini del sistema operativo, scegli Red Hat. • Per Tipo di istanza, scegliete il tipo di istanza più adatto alla vostra PeopleSoft applicazione. Per ulteriori informazioni, consulta Dettagli sull'architettura nella sezione Architettura.	Amministratore del cloud, PeopleSoft amministratore
Installa PeopleSoft sull'istanza.	Installa l' PeopleSoft applicazione e PeopleTools sull'istanza EC2 che hai creato. Per istruzioni, consulta la documentazione Oracle.	Amministratore cloud, PeopleSoft amministratore
Crea il server delle applicazioni.	Crea il server delle applicazioni per il modello AMI e assicurati che si connetta correttamente al database Amazon RDS.	Amministratore cloud, PeopleSoft amministratore
Montare il file system Amazon EFS.	Accedi all'istanza EC2 come utente root ed esegui i seguenti comandi per montare	Amministratore del cloud, PeopleSoft amministratore

Operazione	Description	Competenze richieste

	il file system Amazon EFS in una cartella chiamata PSFTMNT sul server.
	
	`sudo su -`
	`mkdir /psftmnt`
	`cat /etc/fstab`
	
	Aggiungi la riga seguente al file. `/etc/fstab` Usa il nome DNS che hai annotato quando hai creato il file system.
	
	`fs-09e064308f11453`
	`88.efs.us-east-1.a`
	`mazonaws.com:/ /`
	`psftmnt nfs4 nfsvers=4`
	`.1,rsize=1048576,w`
	`size=1048576,hard,`
	`timeo=600,retrans=`
	`2,noresvport,_netdev`
	` 0 0`
	`mount -a`	

Controlla le autorizzazioni.	Assicurati che la PSFTMNT cartella disponga delle autorizzazioni appropriate in modo che l' PeopleSoft utente possa accedervi correttam ente.	Amministratore del cloud, PeopleSoft amministratore

Operazione	Description	Competenze richieste
Crea istanze aggiuntive.	Ripeti i passaggi precedenti di questa epopea per creare istanze modello per il process scheduler, il server web e il server Elasticsearch. Assegna un nome a queste istanze e. `PRCS_TEMPLATE` `WEB_TEMPLATE` `SRCH_TEMPLATE` Per il server Web, imposta `joltPooling=true` e `DynamicConfigReload=1` .	Amministratore cloud, PeopleSoft amministratore

Crea script per configurare i server

Operazione	Description	Competenze richieste

Crea uno script per installare il server delle applicazioni.	Nell'`APP_TEMPLATE` istanza Amazon EC2, come PeopleSoft utente, crea il seguente script. Assegnagli un nome `appstart.sh` e inseriscilo nella `PS_HOME` directory. Utilizzerai questo script per richiamare il server delle applicazioni e registrare anche il nome del server sul mount Amazon EFS.
	`#!/bin/ksh`
	`. /usr/homes/hcmdemo/.profile.`
	`psadmin -c configure -d HCMDEMO`
		PeopleSoft amministratore

Operazione	Description	Competenze richieste
	`psadmin -c parallelb oot -d HCMDEMO touch /psftmnt/`echo $HOSTNAME`	
Crea uno script per installare il server Process Scheduler.	Nell'`PRCS_TEMPLATE` istanza Amazon EC2, come PeopleSoft utente, crea il seguente script. Assegnagl i un nome `prcsstart.sh` e inseriscilo nella `PS_HOME` directory. Utilizzerai questo script per far apparire il server Process Scheduler. `#!/bin/ksh . /usr/homes/hcmdemo/. profile /* The following line ensures that the process scheduler always has a unique name during replaceme nt or scaling activity. */ sed -i "s/.*Prcs ServerName.*/`host name -I | awk -F. '{print "PrcsServ erName=PSUNX"$3$4} '`/" $HOME/appserv/ prcs/*/psprcs.cfg psadmin -p configure -d HCMDEMO psadmin -p start -d HCMDEMO`	PeopleSoft amministratore

Operazione	Description	Competenze richieste

Crea uno script per installare il server Elasticsearch.	Nell'`SRCH_TEMPLATE` istanza Amazon EC2, come utente Elasticsearch, crea lo script seguente. Assegnagli un nome `srchstart.sh` e inseriscilo nella directory `HOME`
	
	`#!/bin/ksh`
	`/* The following line ensures that the correct IP is indicated in the elasticsearch.yaml file. */`
	`sed -i "s/.*network.host.*/`hostname -I
		awk '{print "host:"$0}'`/" $ES_HOME_DIR/config/elasticsearch.yaml`
		`nohup $ES_HOME_DIR/bin/elasticsearch &`
			PeopleSoft amministratore

Operazione	Description	Competenze richieste
Crea uno script per installare il server web.	Nell'`WEB_TEMPLATE` istanza Amazon EC2, come utente del server Web, crea i seguenti script nella directory. `HOME` `renip.sh`: questo script garantisce che il server Web abbia l'IP corretto quando viene clonato dall'AMI. `#!/bin/ksh` `hn=`hostname`` `/* On the following` ` line, change the IP` ` with the hostname with` ` the hostname of the web` ` template. */` `for text_file in `find` ` * -type f -exec grep` ` -l '<hostname-of-the-` `web-template>' {} \;` `do` `sed -e 's/<hostn` `ame-of-the-web-tem` `plate>/'$hn'/g'` ` $text_file > temp` `mv -f temp $text_file` `done`  `psstrsetup.sh` : Questo script garantisce che il server Web utilizzi il server delle applicazioni corretto attualmente in IPs esecuzione. Tenta di connettersi a ciascun server delle applicazioni sulla porta jolt e lo aggiunge al file di configurazione.	PeopleSoft amministratore

Operazione	Description	Competenze richieste
	`#!/bin/ksh` `c2=""` `for ctr in `ls -1 /psftmnt/*.internal` `do` `c1=`echo $ctr | awk -F "/" '{print $3}'` `/* In the following lines, 9000 is the jolt port. Change it if necessary. */` `if nc -z $c1 9000 2> /dev/null; then` `if [[ $c2 = "" ]]; then` `c2="psserver="`echo $c1`":9000"` `else` `c2=`echo $c2`","`echo $c1`":9000"` `fi` `fi` `done` `webstart.sh` : Questo script esegue i due script precedenti e avvia i server Web. `#!/bin/ksh` `/* Change the path in the following if necessary. */` `cd /usr/homes/hcmdemo` `./renip.sh` `./psstrsetup.sh` `webserv/peoplesoft/bin/startPIA.sh`	

Operazione	Description	Competenze richieste
Aggiungi una voce crontab.	Nell'**`WEB_TEMPLATE`** istanza Amazon EC2, come utente del server Web, aggiungi la riga seguente a crontab. Modifica l'ora e il percorso in modo che rispecchino i valori di cui hai bisogno. Questa voce garantisce che il server Web contenga sempre le voci corrette del server delle applicazioni nel `configuration.properties` file. `* * * * * /usr/homes/hcmdemo/psstrsetup.sh`	PeopleSoft amministratore

Creazione AMIs e Auto Scaling di modelli di gruppo

Operazione	Description	Competenze richieste
Crea un AMI per il modello di server delle applicazioni.	Sulla console Amazon EC2, crea un'immagine AMI dell'istanza Amazon `APP_TEMPLATE` EC2. Dai un nome all'AMI`PSAPPSRV-SCG-VER1`. Per istruzioni, consulta la documentazione di Amazon EC2.	Amministratore cloud, PeopleSoft amministratore
Crea AMIs per gli altri server.	Ripeti il passaggio precedente AMIs per creare per lo scheduler di processo, il server Elasticsearch e il server web.	Amministratore del cloud, amministratore PeopleSoft

Operazione	Description	Competenze richieste

Crea un modello di avvio per il gruppo Auto Scaling del server di applicazioni.	Crea un modello di avvio per il gruppo Auto Scaling del server di applicazioni. Assegna un nome al modello `PSAPPSRV_TEMPLATE`. Nel modello, scegli l'AMI che hai creato per l'`APP_TEMPLATE` istanza. Per istruzioni, consulta la documentazione di Amazon EC2.
	• Nel modello di lancio, seleziona il tipo di istanza in base ai tuoi requisiti.
	• Nel campo Dati utente della sezione Dettagli avanzati, aggiungi le seguenti voci. Assicurati che il percorso e le informazioni sull'utente siano corretti. Hai creato lo `appstart.sh` script in un passaggio precedente.
	`#! /bin/ksh`
	`su -c "/usr/homes/hcmdemo/appstart.sh" - hcmdemo`
		Amministratore cloud, PeopleSoft amministratore

Operazione	Description	Competenze richieste

Crea un modello di avvio per il gruppo Auto Scaling del server Process Scheduler.	Ripetere il passaggio precedente per creare un modello di avvio per il gruppo Auto Scaling del server Process Scheduler. Assegna un nome al modello. `PSPRCS_TEMPLATE` Nel modello, scegli l'AMI che hai creato per lo scheduler dei processi.
	• Nel campo Dati utente della sezione Dettagli avanzati, aggiungi le seguenti voci. Assicurati che il percorso e le informazioni sull'utente siano corretti. Hai creato lo `prcsstart.sh` script in un passaggio precedente.
	`#! /bin/ksh`
	`su -c "/usr/hom`
	`es/hcmdemo/prcssta`
	`rt.sh" - hcmdemo`
		Amministratore cloud, PeopleSoft amministratore

Operazione	Description	Competenze richieste

Crea un modello di lancio per il gruppo Auto Scaling del server Elasticsearch.	Ripeti i passaggi precedenti per creare un modello di avvio per il gruppo Auto Scaling del server Elasticsearch. Assegna un nome al modello. `SRCH_TEMPLATE` Nel modello, scegli l'AMI che hai creato per il server di ricerca.
	• Nel campo Dati utente della sezione Dettagli avanzati, aggiungi le seguenti voci. Assicurati che il percorso e le informazioni sull'utente siano corretti. Hai creato lo `srchstart.sh` script in un passaggio precedente.
	`#! /bin/ksh`
	`su -c "/usr/homes/essearch/srchstart.sh" - essearch`
		Amministratore cloud, PeopleSoft amministratore

Operazione	Description	Competenze richieste

Crea un modello di avvio per il gruppo Auto Scaling del server web.	Ripetere i passaggi precedenti per creare un modello di avvio per il gruppo Auto Scaling del server Web. Assegna un nome al modello`WEB_TEMPLATE` . Nel modello, scegli l'AMI che hai creato per il server web.
	• Nel campo Dati utente della sezione Dettagli avanzati, aggiungi le seguenti voci. Assicurati che il percorso e le informazioni sull'utente siano corretti. Hai creato lo `webstart.sh` script in un passaggio precedente.
	`#! /bin/ksh`
	`su -c "/usr/homes/hcmdemo/webstart.sh" - hcmdemo`
		Amministratore cloud, PeopleSoft amministratore

Creare gruppi di Auto Scaling

Operazione	Description	Competenze richieste
Crea un gruppo Auto Scaling per il server delle applicazioni.	Sulla console Amazon EC2, crea un gruppo Auto Scaling `PSAPPSRV_ASG` chiamato per il server delle applicazioni utilizzando il modello. `PSAPPSRV_TEMPLATE` Per istruzioni, consulta la	Amministratore del cloud, amministratore PeopleSoft

Operazione	Description	Competenze richieste
	documentazione di Amazon EC2. • Nella pagina Scegli le opzioni di avvio dell'istanza, seleziona il VPC corretto, quindi seleziona più sottoreti da diverse zone di disponibilità. • Nella pagina Configura opzioni avanzate, non selezionare un sistema di bilanciamento del carico. • Nella pagina Configura la dimensione del gruppo e le politiche di scalabilità, scegli le impostazioni in base alla quantità di carico per cui desideri progettare il sistema e se desideri utilizzare una politica di scalabilità. Ti consigliamo di impostare la capacità minima e desiderata su almeno 2 in modo che sia disponibile almeno un'istanza per gestire il traffico in qualsiasi momento. Per ulteriori informazioni sulle politiche di Auto Scaling, consulta la documentazione di Amazon EC2.	

Operazione	Description	Competenze richieste
Crea gruppi di Auto Scaling per gli altri server.	Ripetere il passaggio precedente per creare gruppi di Auto Scaling per il process scheduler, il server Elasticse arch e il server web.	Amministratore del cloud, amministratore PeopleSoft

Crea e configura gruppi target

Operazione	Description	Competenze richieste
Crea un gruppo target per il server web.	Sulla console Amazon EC2, crea un gruppo target per il server Web. Per istruzioni, consulta la documentazione di Elastic Load Balancing. Imposta la porta sulla porta su cui il server Web è in ascolto.	Amministratore cloud
Configura i controlli sanitari.	Verifica che i controlli sanitari abbiano i valori corretti per riflettere i tuoi requisiti aziendali. Per ulteriori informazioni, consulta la Guida per l'utente di Elastic Load Balancing.	Amministratore cloud
Crea un gruppo target per il server Elasticsearch.	Ripeti i passaggi precedent i per creare un gruppo target chiamato `PSFTSRCH` per il server Elasticsearch e imposta la porta Elasticsearch corretta.	Amministratore cloud
Aggiungi gruppi target ai gruppi di Auto Scaling.	Apri il gruppo Auto Scaling del server web chiamato `PSPIA_ASG` che hai creato	Amministratore cloud

Operazione	Description	Competenze richieste
	in precedenza. Nella scheda Bilanciamento del carico, scegli Modifica, quindi aggiungi il gruppo PSFTWEB target al gruppo Auto Scaling. Ripeti questo passaggio per il gruppo Elasticsearch Auto Scaling per aggiungere il PSSRCH_ASG PSFTSRCH gruppo target creato in precedenza.	
Imposta la persistenza della sessione.	Nel gruppo targetPSFTWEB, scegli la scheda Attributi, scegli Modifica e imposta la persistenza della sessione. Per il tipo di adesività, scegli Load Balancer generated cookie e imposta la durata su 1. Per ulteriori informazioni, consulta la Guida per l'utente di Elastic Load Balancing. Ripeti questo passaggio per il gruppo target. PSFTSRCH	Amministratore cloud

Crea e configura i sistemi di bilanciamento del carico delle applicazioni

Operazione	Description	Competenze richieste
Crea un sistema di bilanciam ento del carico per i server Web.	Crea un Application Load Balancer denominato PSFTLB per bilanciare il carico del traffico verso i server Web.	Amministratore cloud

Operazione	Description	Competenze richieste
	Per istruzioni, consulta la documentazione di Elastic Load Balancing. • Fornisci il nome del load balancer. • Per Scheme (Schema), scegliere Internet-facing. • Nella sezione Mappatura della rete, seleziona il VPC corretto e almeno due sottoreti pubbliche da diverse zone di disponibilità. • Nella sezione Listener and routing, seleziona il gruppo di destinazione e specifica il protocollo PSFTWEB e il numero di porta corretti.	

Operazione	Description	Competenze richieste
Crea un sistema di bilanciam ento del carico per i server Elasticsearch.	Crea un Application Load Balancer denominato PSFTSCH per bilanciare il carico del traffico verso i server Elasticsearch. • Fornisci il nome del load balancer. • Per Schema, scegli Interno. • Nella sezione Mappatura della rete, seleziona il VPC e le sottoreti private corrette. • Nella sezione Listener and routing, seleziona il gruppo di destinazione e specifica il protocollo PSFTSRCH e il numero di porta corretti.	Amministratore cloud
Configura Route 53.	Sulla console Amazon Route 53, crea un record nella zona ospitata che servirà l' PeopleSoft applicazione. Per istruzioni, consulta la documentazione di Amazon Route 53. Ciò garantisce che tutto il traffico passi attraverso il sistema di PSFTLB bilanciam ento del carico.	Amministratore cloud

Risorse correlate

- PeopleSoft Sito web Oracle
- Documentazione AWS

Configura il disaster recovery per Oracle JD Edwards con EnterpriseOne AWS Elastic Disaster Recovery

Thanigaivel Thirumalai, Amazon Web Services

Riepilogo

I disastri causati da catastrofi naturali, guasti delle applicazioni o interruzioni dei servizi danneggiano i ricavi e causano interruzioni delle applicazioni aziendali. Per ridurre le ripercussioni di tali eventi, la pianificazione del disaster recovery (DR) è fondamentale per le aziende che adottano i sistemi ERP (EnterpriseOne Enterprise Resource Planning) di JD Edwards e altri software mission critical e aziendali.

Questo modello spiega come le aziende possono utilizzare AWS Elastic Disaster Recovery come opzione di DR per le loro applicazioni JD Edwards. EnterpriseOne Descrive inoltre i passaggi per utilizzare il failover e il failback di Elastic Disaster Recovery per creare una strategia di DR interregionale per i database ospitati su un'istanza Amazon Elastic Compute Cloud (Amazon EC2) nel cloud AWS.

> **Note**
>
> Questo modello richiede che le regioni primarie e secondarie per l'implementazione del DR tra regioni siano ospitate su AWS.

Oracle JD Edwards EnterpriseOne è una soluzione software ERP integrata per aziende di medie e grandi dimensioni in un'ampia gamma di settori.

AWS Elastic Disaster Recovery riduce al minimo i tempi di inattività e la perdita di dati con un ripristino rapido e affidabile di applicazioni locali e basate sul cloud utilizzando storage conveniente, elaborazione e ripristino minimi. point-in-time

AWS fornisce quattro modelli di architettura DR di base. Questo documento si concentra su configurazione, configurazione e ottimizzazione utilizzando la strategia Pilot Light. Questa strategia consente di creare un ambiente di ripristino di emergenza a basso costo in cui si fornisce inizialmente un server di replica per la replica dei dati dal database di origine e il provisioning del server di database effettivo solo quando si avvia un'operazione di drill and recovery. Questa strategia elimina i costi di manutenzione di un server di database nella regione DR. Invece, paghi per un'istanza EC2 più piccola che funge da server di replica.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un' EnterpriseOne applicazione JD Edwards in esecuzione su Oracle Database o Microsoft SQL Server con un database supportato in uno stato di esecuzione su un'istanza EC2 gestita. Questa applicazione deve includere tutti i componenti di EnterpriseOne base di JD Edwards (Enterprise Server, HTML Server e Database Server) installati in una regione AWS.
- Un ruolo AWS Identity and Access Management (IAM) per configurare il servizio Elastic Disaster Recovery.
- La rete per l'esecuzione di Elastic Disaster Recovery è configurata in base alle impostazioni di connettività richieste.

Limitazioni

- Puoi utilizzare questo modello per replicare tutti i livelli, a meno che il database non sia ospitato su Amazon Relational Database Service (Amazon RDS), nel qual caso ti consigliamo di utilizzare la funzionalità di copia interregionale di Amazon RDS.
- Elastic Disaster Recovery non è compatibile con CloudEndure Disaster Recovery, ma puoi eseguire l'aggiornamento da Disaster Recovery. CloudEndure Per ulteriori informazioni, consulta le domande frequenti nella documentazione di Elastic Disaster Recovery.
- Amazon Elastic Block Store (Amazon EBS) limita la velocità con cui è possibile scattare istantanee. Puoi replicare un numero massimo di 300 server in un singolo account AWS utilizzando Elastic Disaster Recovery. Per replicare più server, puoi utilizzare più account AWS o più regioni AWS di destinazione. (Dovrai configurare Elastic Disaster Recovery separatamente per ogni account e regione). Per ulteriori informazioni, consulta le best practice nella documentazione di Elastic Disaster Recovery.
- I carichi di lavoro di origine (l' EnterpriseOne applicazione e il database JD Edwards) devono essere ospitati su istanze EC2. Questo modello non supporta carichi di lavoro on-premise o in altri ambienti cloud.
- Questo modello si concentra sui componenti JD EnterpriseOne Edwards. Un piano completo di disaster recovery e business continuity (BCP) dovrebbe includere altri servizi di base, tra cui:
  - Rete (cloud privato virtuale, sottoreti e gruppi di sicurezza)
  - Active Directory

- Amazon WorkSpaces
- Elastic Load Balancing
- Un servizio di database gestito come Amazon Relational Database Service (Amazon RDS)

Per ulteriori informazioni su prerequisiti, configurazioni e limitazioni, consulta la documentazione di Elastic Disaster Recovery.

Versioni del prodotto

- Oracle JD Edwards EnterpriseOne (versioni supportate da Oracle e SQL Server basate sui requisiti tecnici minimi di Oracle)

Architecture

Stack tecnologico Target

- Un'unica regione e un singolo cloud privato virtuale (VPC) per la produzione e la non produzione e una seconda regione per il DR
- Zone di disponibilità singole per garantire una bassa latenza tra i server
- Un Application Load Balancer che distribuisce il traffico di rete per migliorare la scalabilità e la disponibilità delle applicazioni su più zone di disponibilità
- Amazon Route 53 per fornire la configurazione DNS (Domain Name System)
- Amazon fornirà WorkSpaces agli utenti un'esperienza desktop nel cloud
- Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) per l'archiviazione di backup, file e oggetti
- Amazon CloudWatch per la registrazione, il monitoraggio e gli allarmi delle applicazioni
- Amazon Elastic Disaster Recovery per il disaster recovery

Architettura di destinazione

Il diagramma seguente mostra l'architettura di disaster recovery interregionale per JD Edwards che EnterpriseOne utilizza Elastic Disaster Recovery.

Procedura

Ecco una revisione di alto livello del processo. Per i dettagli, consulta la sezione Epics.

- La replica di Elastic Disaster Recovery inizia con una sincronizzazione iniziale. Durante la sincronizzazione iniziale, AWS Replication Agent replica tutti i dati dai dischi di origine alla risorsa appropriata nella sottorete dell'area di staging.
- La replica continua continua a tempo indeterminato dopo il completamento della sincronizzazione iniziale.
- Dopo l'installazione dell'agente e l'avvio della replica, si esaminano i parametri di avvio, che includono configurazioni specifiche del servizio e un modello di lancio di Amazon EC2. Quando il server di origine viene indicato come pronto per il ripristino, puoi avviare le istanze.
- Quando Elastic Disaster Recovery emette una serie di chiamate API per iniziare l'operazione di avvio, l'istanza di ripristino viene immediatamente avviata su AWS in base alle impostazioni di avvio. Il servizio attiva automaticamente un server di conversione durante l'avvio.
- La nuova istanza viene avviata su AWS al termine della conversione ed è pronta per l'uso. Lo stato del server di origine al momento del lancio è rappresentato dai volumi associati all'istanza lanciata. Il processo di conversione prevede modifiche ai driver, alla rete e alla licenza del sistema operativo per garantire che l'istanza si avvii nativamente su AWS.
- Dopo il lancio, i volumi appena creati non vengono più sincronizzati con i server di origine. AWS Replication Agent continua a replicare regolarmente le modifiche apportate ai server di origine nei volumi dell'area di staging, ma le istanze avviate non riflettono tali modifiche.
- Quando avvii una nuova istanza drill o recovery, i dati si riflettono sempre nello stato più recente che è stato replicato dal server di origine alla sottorete dell'area di staging.
- Quando il server di origine è contrassegnato come pronto per il ripristino, è possibile avviare le istanze.

> **Note**
>
> Il processo funziona in entrambi i modi: per il failover da una regione AWS primaria a una regione DR e per il failback sul sito primario, una volta ripristinato. Puoi prepararti al failback invertendo la direzione della replica dei dati dal computer di destinazione al computer di origine in modo completamente orchestrato.

I vantaggi di questo processo descritto in questo modello includono:

- Flessibilità: i server di replica sono scalabili e scalabili in base al set di dati e al tempo di replica, in modo da poter eseguire test di DR senza interrompere i carichi di lavoro o la replica di origine.
- Affidabilità: la replica è solida, senza interruzioni e continua.
- Automazione: questa soluzione fornisce un processo unificato e automatizzato per test, ripristino e failback.
- Ottimizzazione dei costi: è possibile replicare solo i volumi necessari e pagarli, e pagare le risorse di elaborazione presso il sito DR solo quando tali risorse vengono attivate. È possibile utilizzare un'istanza di replica ottimizzata in termini di costi (si consiglia di utilizzare un tipo di istanza ottimizzata per il calcolo) per più fonti o un'unica fonte con un volume EBS di grandi dimensioni.

Automazione e scalabilità

Quando si esegue il disaster recovery su larga scala, i EnterpriseOne server JD Edwards dipenderanno da altri server dell'ambiente. Esempio:

- Gli EnterpriseOne application server JD Edwards che si connettono a un database EnterpriseOne supportato da JD Edwards all'avvio hanno dipendenze da quel database.
- EnterpriseOne I server JD Edwards che richiedono l'autenticazione e devono connettersi a un controller di dominio all'avvio per avviare i servizi dipendono dal controller di dominio.

Per questo motivo, si consiglia di automatizzare le attività di failover. Ad esempio, puoi utilizzare AWS Lambda o AWS Step Functions per automatizzare gli script di EnterpriseOne avvio di JD Edwards e le modifiche al bilanciamento del carico per automatizzare il processo di failover. end-to-end Per ulteriori informazioni, consulta il post sul blog Creazione di un piano di disaster recovery scalabile con AWS Elastic Disaster Recovery.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Block Store (Amazon EBS) fornisce volumi di archiviazione a livello di blocchi da utilizzare con le istanze EC2.
- Amazon Elastic Compute Cloud (Amazon EC2) Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.

- AWS Elastic Disaster Recovery riduce al minimo i tempi di inattività e la perdita di dati con un ripristino rapido e affidabile di applicazioni locali e basate sul cloud utilizzando storage conveniente, elaborazione e ripristino minimi. point-in-time
- Amazon Virtual Private Cloud (Amazon VPC) ti offre il pieno controllo del tuo ambiente di rete virtuale, inclusi posizionamento delle risorse, connettività e sicurezza.

Best practice

Le migliori pratiche generali

- Prepara un piano scritto su cosa fare in caso di un vero evento di recupero.
- Dopo aver configurato correttamente Elastic Disaster Recovery, crea un CloudFormation modello AWS in grado di creare la configurazione su richiesta, in caso di necessità. Determina l'ordine in cui i server e le applicazioni devono essere avviati e registralo nel piano di ripristino.
- Esegui un'analisi regolare (si applicano le tariffe standard di Amazon EC2).
- Monitora lo stato della replica in corso utilizzando la console Elastic Disaster Recovery o a livello di programmazione.
- Proteggi le point-in-time istantanee e conferma prima di chiudere le istanze.
- Crea un ruolo IAM per l'installazione di AWS Replication Agent.
- Abilita la protezione dalla terminazione per le istanze di ripristino in uno scenario di DR reale.
- Non utilizzare l'azione Disconnect from AWS nella console Elastic Disaster Recovery per i server per i quali hai avviato le istanze di ripristino, anche nel caso di un evento di ripristino reale. L'esecuzione di una disconnessione interrompe tutte le risorse di replica relative a questi server di origine, inclusi i punti di ripristino point-in-time (PIT).
- Modificate la policy PIT per modificare il numero di giorni per la conservazione delle istantanee.
- Modifica il modello di avvio nelle impostazioni di avvio di Elastic Disaster Recovery per impostare la sottorete, il gruppo di sicurezza e il tipo di istanza corretti per il server di destinazione.
- Automatizza il processo di end-to-end failover utilizzando Lambda o Step Functions per automatizzare gli script di avvio di JD Edwards EnterpriseOne e le modifiche al load balancer.

EnterpriseOne Ottimizzazione e considerazioni su JD Edwards

- Passa al PrintQueuedatabase.
- MediaObjectsSpostatevi nel database.

- Escludi i log e la cartella temporanea dai server batch e logici.
- Escludi la cartella temporanea da Oracle. WebLogic
- Crea script per l'avvio dopo il failover.
- Escludere il tempdb per SQL Server.
- Escludi il file temporaneo per Oracle.

Epiche

Esegui le attività e la configurazione iniziali

Operazione	Description	Competenze richieste
Configurare la rete di replica.	Implementa il tuo Enterpris eOne sistema JD Edwards nella regione AWS principale e identifica la regione AWS per il DR. Segui i passaggi nella sezione Requisiti della rete di replica della documentazione di Elastic Disaster Recovery per pianificare e configurare la tua rete di replica e DR.	Amministratore AWS
Determina RPO e RTO.	Identifica il Recovery Time Objective (RTO) e il Recovery Point Objective (RPO) per i server delle applicazioni e il database.	Architetto cloud, architetto DR
Abilita la replica per Amazon EFS.	Se applicabile, abilita la replica dalla regione AWS primaria a quella DR per file system condivisi come Amazon Elastic File System (Amazon EFS) utilizzando	Amministratore del cloud

Operazione	Description	Competenze richieste
	AWS DataSync, rsync o un altro strumento appropriato.	
Gestisci il DNS in caso di DR.	Identifica il processo di aggiornamento del Domain Name System (DNS) durante l'esercitazione di DR o il DR. effettivo	Amministratore cloud
Crea un ruolo IAM per la configurazione.	Segui le istruzioni nella sezione Inizializzazione e autorizzazioni di Elastic Disaster Recovery della documentazione di Elastic Disaster Recovery per creare un ruolo IAM per inizializzare e gestire il servizio AWS.	Amministratore del cloud
Configura il peering VPC.	Assicurati che l'origine e la destinazione VPCs siano peer e accessibili l'una all'altra. Per istruzioni di configurazione, consulta la documentazione di Amazon VPC.	Amministratore AWS

Configura le impostazioni di replica di Elastic Disaster Recovery

Operazione	Description	Competenze richieste
Inizializza Elastic Disaster Recovery.	Apri la console Elastic Disaster Recovery, scegli la regione AWS di destinazi one (dove replicherai i dati e lancerai le istanze di ripristin o), quindi scegli Imposta	Amministratore AWS

Operazione	Description	Competenze richieste
	impostazioni di replica predefinite.	
Configura i server di replica.	1. Nel riquadro Configura i server di replica, inserisci la sottorete dell'area di gestione temporanea e il tipo di istanza del server di replica. Il tipo di istanza `t3.small` è selezionato di default. Configura questa impostazione in base ai tuoi requisiti e ricorda di considerare i prezzi delle istanze. Per ulteriori informazioni, consulta Prezzi di Amazon EC2. 2. Nella sezione Accesso al servizio, scegli Visualizza dettagli per esaminare il ruolo collegato al servizio e le politiche aggiuntive create durante l'inizializzazione del servizio. 3. Scegli Next (Successivo).	Amministratore AWS

Operazione	Description	Competenze richieste
Configura volumi e gruppi di sicurezza.	1. Nel riquadro Volumi e gruppi di sicurezza, seleziona il tipo di volume EBS per il server di replica e imposta la crittografia Amazon EBS su Default. 2. Seleziona Usa sempre il gruppo di sicurezza AWS Elastic Disaster Recovery in modo che Elastic Disaster Recovery colleghi e monitori automaticamente il gruppo di sicurezza predefinito. 3. Scegli Next (Successivo).	Amministratore AWS

Operazione	Description	Competenze richieste
Configura impostazioni aggiuntive.	1. Nel riquadro Impostazioni aggiuntive, configura il routing e la limitazione dei dati, la politica PIT e i tag. • Il routing e la limitazione dei dati controllano il modo in cui i dati fluiscono dal server esterno ai server di replica. Scegli Usa IP privato per la replica dei dati. In caso contrario, ai server di replica verrà assegnato automaticamente un IP pubblico e i dati fluiranno sulla rete Internet pubblica. • Nella sezione Point in time (PIT), configura una politica di conservazione che determini la durata dopo la quale non sono necessarie le istantanee. Il periodo di conservazione predefinito è 60 giorni. • Nella sezione Tag, aggiungi tag personalizzati alle risorse create da Elastic Disaster Recovery nel tuo account AWS. 2. Scegli Avanti, rivedi le impostazioni nel riquadro	Amministratore AWS

Operazione	Description	Competenze richieste
	successivo, quindi scegli Crea default per creare il modello predefinito.	

Installa l'agente di replica AWS

Operazione	Description	Competenze richieste
Crea un ruolo IAM.	Crea un ruolo IAM che contenga la `AWSElasticDisasterRecoveryAgentInstallationPolicy` policy. Nella sezione Seleziona il tipo di accesso AWS, abilita l'accesso programmatico. Annota l'ID della chiave di accesso e la chiave di accesso segreta. Queste informazioni ti serviranno durante l'installazione di AWS Replication Agent.	Amministratore AWS
Verifica i requisiti.	Verifica e completa i prerequisiti nella documentazione di Elastic Disaster Recovery per l'installazione di AWS Replication Agent.	Amministratore AWS
Installa l'agente di replica AWS.	Segui le istruzioni di installazione per il tuo sistema operativo e installa AWS Replication Agent.	Amministratore AWS

Operazione	Description	Competenze richieste

	• Per Microsoft Windows: scarica i file di installazione ed esegui il file.exe come amministratore. Rispondi alle istruzioni per completar e l'installazione.
	• Per Linux: copia i seguenti comandi (nell'ordine presentato) e incollali nella sessione Secure Shell (SSH). Il primo comando scarica il programma di installazione e il secondo lo esegue.
	`wget -O ./aws-replication-installer-init.py https://aws-elastic-disaster-recovery-us-west-2.s3.amazonaws.com/latest/linux/aws-replication-installer-init.py`
	`sudo python3 aws-replication-installer-init.py`
	Rispondi alle istruzioni per completare l'installazione.	

Operazione	Description	Competenze richieste
	Note Modifica l'URL in modo che rifletta la tua regione. Ripeti questi passaggi per il server rimanente.	
Monitora la replica.	Torna al riquadro Elastic Disaster Recovery Source servers per monitorare lo stato della replica. La sincroniz zazione iniziale richiederà del tempo a seconda delle dimensioni del trasferimento dei dati. Quando il server di origine è completamente sincroniz zato, lo stato del server verrà aggiornato a Ready. Ciò significa che è stato creato un server di replica nell'area di gestione temporanea e che i volumi EBS sono stati replicati dal server di origine all'area di gestione temporanea.	Amministratore AWS

Configura le impostazioni di avvio

Operazione	Description	Competenze richieste
Modifica le impostazioni di avvio.	Per aggiornare le impostazioni di avvio per le istanze di drill e recovery, nella console Elastic Disaster Recovery, seleziona il server di origine, quindi scegli Azioni, Modifica impostazioni di avvio. In alternativa, puoi scegliere le macchine di origine che si replicano dalla pagina Server di origine, quindi scegliere la scheda Impostazioni di avvio. Questa scheda è composta da due sezioni: Impostazioni generali di avvio e Modello di avvio EC2.	Amministratore AWS
Configura le impostazioni generali di avvio.	Modifica le impostazioni generali di avvio in base alle tue esigenze. • Dimensionamento corretto del tipo di istanza: se scegli Basic, Elastic Disaster Recovery ignora il tipo di istanza selezionato nel modello di lancio di Amazon EC2 e sceglie automaticamente il tipo di istanza in base al sistema operativo, alla CPU e alla RAM del server di origine.	Amministratore AWS

Operazione	Description	Competenze richieste
	• Copia l'IP privato: scegli se desideri che Elastic Disaster Recovery assicuri che l'IP privato utilizzato dal drill o dall'istanza di ripristino corrisponda all'IP privato utilizzato dal server di origine. Se hai scelto Sì, assicurati che l'intervallo IP della sottorete impostato nel modello di lancio di Amazon EC2 includa l'indirizzo IP privato. Per ulteriori informazioni, consulta Impostazioni generali di avvio nella documentazione di Elastic Disaster Recovery.	

Operazione	Description	Competenze richieste
Configura il modello di lancio di Amazon EC2.	Elastic Disaster Recovery utilizza i modelli di lancio di Amazon EC2 per avviare istanze di drill and recovery per ogni server di origine. Il modello di lancio viene creato automaticamente per ogni server di origine che aggiungi a Elastic Disaster Recovery dopo l'installazione di AWS Replication Agent. È necessario impostare il modello di lancio di Amazon EC2 come modello di avvio predefinito se si desidera utilizzarlo con Elastic Disaster Recovery. Per ulteriori informazioni, consulta EC2 Launch Template nella documentazione di Elastic Disaster Recovery.	Amministratore AWS

Avvia il drill and failover del DR

Operazione	Description	Competenze richieste
Avvia Drill	1. Sulla console Elastic Disaster Recovery, apri la pagina Server di origine e verifica che lo stato del server di origine sia Pronto.	Amministratore AWS

Operazione	Description	Competenze richieste
	2. Seleziona tutti i server di origine per i quali desideri eseguire l'esercitazione DR. 3. Dal menu Initiate recovery job, scegli Initiate drill e seleziona l'istantanea appropriata. point-in-time Questo avvia un processo di ripristino per i server di origine selezionati. È possibile monitorare lo stato del processo nella scheda Cronologia del processo di ripristino. L'istanza drill lanciata appare anche nella pagina delle istanze di ripristino. **Note** Le ulteriori modifiche al server di origine verranno sincronizzate con il server di replica, non con l'istanza drill. 4. Testa e verifica l'istanza DR drill. 5. Nella pagina Recovery Instances, seleziona l'istanza drill, quindi scegli Actions, Disconnect from	

Operazione	Description	Competenze richieste
	AWS. Ciò elimina l'agente AWS Replication dall'istanza di ripristino e rimuove tutte le risorse associate all'istanza di ripristino da Elastic Disaster Recovery. 6. Scegli Elimina istanze di ripristino. Ciò elimina la rappresentazione dell'istanza dalla console Elastic Disaster Recovery e dissocia completamente l'istanza dal servizio Elastic Disaster Recovery. Non elimina l'istanza EC2 sottostante. 7. Termina l'istanza DR drill dalla console Amazon EC2. Per ulteriori informazioni, consulta Preparazione per il failover nella documentazione di Elastic Disaster Recovery.	

Operazione	Description	Competenze richieste
Convalida l'esercitazione.	Nel passaggio precedente, hai lanciato nuove istanze target nella regione DR. Le istanze di destinazione sono repliche dei server di origine basate sull'istantanea scattata al momento dell'avvio. In questa procedura, ti connetti ai tuoi computer di destinazione Amazon EC2 per confermare che funzionino come previsto. 1. Aprire la console di Amazon EC2. 2. Scegli Istanze (in esecuzione). 3. Seleziona l'istanza di destinazione e annota il suo IPv4 indirizzo privato. 4. Assicurati di poterti connettere all'istanza EC2 e che JD Edwards EnterpriseOne e i relativi componenti vengano replicati come previsto.	

Operazione	Description	Competenze richieste
Avvia un failover.	Un failover è il reindiriz zamento del traffico da un sistema primario a un sistema secondario. Elastic Disaster Recovery ti aiuta a eseguire un failover avviando istanze di ripristino su AWS. Una volta avviate le istanze di ripristin o, il traffico dai sistemi primari viene reindirizzato a queste istanze. 1. Nella console Elastic Disaster Recovery, apri la pagina Server di origine e verifica che la colonna Pronto per il ripristino per il server di origine indichi Ready e che la colonna Data replication status indichi Healthy. 2. Seleziona il server di origine. Dal menu Avvia processo di ripristino, scegli Avvia ripristino. 3. Seleziona l' point-in-timeistantanea da cui avviare l'istanza di ripristino, quindi scegli Avvia ripristin o. Questo avvia un processo di ripristino. È possibile monitorare lo stato del	Amministratore AWS

Operazione	Description	Competenze richieste
	processo nella pagina Istanze di ripristino. 4. Testa e verifica l'istanza di ripristino. Se necessario, modifica la configurazione DNS e collega l' Enterpris eOne applicazione JD Edwards al database. 5. È ora possibile disconnet tere e disattivare il Enterpris eOne server JD Edwards di origine, poiché tutte le modifiche sono state scritte nella nuova istanza di ripristino. 6. Registra l'istanza di ripristin o come server di origine nella regione DR seguendo la procedura descritta nell'epic Install the AWS Replication Agent. Per ulteriori informazioni, consulta Esecuzione di un failover nella documentazione di Elastic Disaster Recovery.	

Operazione	Description	Competenze richieste
Avvia un failback.	Il processo di avvio di un failback è simile al processo di avvio del failover. 1. Apri la console Elastic Disaster Recovery nella regione principale. Vai alla pagina delle istanze di ripristino, seleziona l'istanza drill, quindi scegli Actions, Disconnect from AWS, Delete recovery instances. 2. Apri la console Elastic Disaster Recovery nella regione DR. Registra il tuo nuovo server JD Edwards come EnterpriseOne server di origine nella regione DR installando AWS Replicati on Agent. I dati verranno sincronizzati con un nuovo server di replica fornito nella nuova sottorete di staging. **Note** Quando il nuovo server JD Edwards viene registrato come Enterpris eOne server di origine, nella console Elastic Disaster Recovery potrebbero essere	Amministratore AWS

Operazione	Description	Competenze richieste
	presenti due server di origine: un server creato dall'ista nza EC2 primaria e il nuovo server creato dall'ista nza di ripristin o. Ti consiglia mo di etichettare correttamente i server per evitare confusione e preferibilmente di aggiungere il nuovo server al modello di lancio. 3. Per riavviare la replica DR dalla regione primaria, dissocia l'istanza di ripristin o lanciata dalla console Elastic Disaster Recovery nella regione DR e registra l'host come server di origine nella regione primaria. Per ulteriori informazioni, consulta Esecuzione di un failback nella documentazione di Elastic Disaster Recovery.	

Operazione	Description	Competenze richieste
Avvia i componenti di JD Edwards. EnterpriseOne	1. Avvia il database JD Edwards accedendo al server del EnterpriseOne database. 2. Quando il database è in esecuzione, avvia i server logici e batch di JD Edwards. EnterpriseOne 3. Avviare WebLogic sui server Web e avviare un'istanza JAS sui server JAS. 4. Avviare WebLogic sul server di provisioning e sul server per la console SM. 5. Avviare SM Agent sui server. 6. Conferma che l'accesso a JD Edwards EnterpriseOne funzioni correttamente. È necessario incorporare le modifiche in Route 53 e Application Load Balancer affinché il collegamento JD Edwards funzioni. Enterpris eOne È possibile automatizzare questi passaggi utilizzan do Lambda, Step Functions e Systems Manager (Run Command).	J.D. Edwards CNC Enterpris eOne

Operazione	Description	Competenze richieste
	Note Elastic Disaster Recovery esegue la replica a livello di blocco dei volumi EBS dell'istanza EC2 di origine che ospitano il sistema operativo e i file system. I file system condivisi creati utilizzando Amazon EFS non fanno parte di questa replica. Puoi replicare i file system condivisi nella regione DR utilizzando AWS DataSync, come indicato nella prima epic, e poi montare questi file system replicati nel sistema DR.	

Risoluzione dei problemi

Problema	Soluzione
Lo stato di replica dei dati del server di origine è in fase di stallo e la replica è in ritardo. Se si controllano i dettagli, lo stato di replica dei dati visualizza Agente non visualizzato.	Verificare che il server di origine bloccato sia in esecuzione.

Problema	Soluzione
	Note Se il server di origine non funziona, il server di replica viene chiuso automatic amente. Per ulteriori informazioni sui problemi di ritardo, consulta Problemi di ritardo nella replica nella documentazione di Elastic Disaster Recovery.
L'installazione di AWS Replication Agent nell'istanza EC2 di origine non riesce in RHEL 8.2 dopo la scansione dei dischi. `aws_replication_agent_installer.log` rivela che mancano gli header del kernel.	Prima di installare AWS Replication Agent su RHEL 8, CentOS 8 o Oracle Linux 8, esegui: `sudo yum install elfutils-libelf-devel` Per ulteriori informazioni, consulta i requisiti di installazione di Linux nella documentazione di Elastic Disaster Recovery.
Nella console Elastic Disaster Recovery, il server di origine viene visualizzato come Pronto con un ritardo e lo stato di replica dei dati è impostato su Stallato. A seconda di quanto tempo l'AWS Replicati on Agent non è disponibile, lo stato potrebbe indicare un ritardo elevato, ma il problema rimane lo stesso.	Utilizza un comando del sistema operativo per confermare che l'agente AWS Replication è in esecuzione nell'istanza EC2 di origine o conferma che l'istanza è in esecuzione. Dopo aver corretto eventuali problemi, Elastic Disaster Recovery riavvierà la scansione. Attendi che tutti i dati siano stati sincronizzati e che lo stato della replica sia integro prima di iniziare un'esercitazione di DR.

Problema	Soluzione
Replica iniziale con elevato ritardo. Sulla console Elastic Disaster Recovery, puoi vedere che lo stato di sincronizzazione iniziale è estremamente lento per un server di origine.	Verifica i problemi di ritardo di replica documentati nella sezione Problemi di ritardo di replica della documentazione di Elastic Disaster Recovery. Il server di replica potrebbe non essere in grado di gestire il carico a causa di operazioni di elaborazione intrinseche. In tal caso, prova ad aggiornare il tipo di istanza dopo aver consultato il team di AWS Technical Support.

Risorse correlate

- Guida per l'utente di AWS Elastic Disaster Recovery
- Creazione di un piano di disaster recovery scalabile con AWS Elastic Disaster Recovery (post sul blog AWS)
- AWS Elastic Disaster Recovery: un'introduzione tecnica (corso AWS Skill Builder; richiede l'accesso)
- Guida introduttiva rapida di AWS Elastic Disaster Recovery

Imposta il rilevamento della CloudFormation deriva in un'organizzazione multiregionale e con più account

Ram Kandaswamy, Amazon Web Services

Riepilogo

Gli utenti di Amazon Web Services (AWS) spesso cercano un modo efficiente per rilevare le discrepanze nella configurazione delle risorse, tra cui la deriva negli AWS CloudFormation stack, e correggerle il prima possibile. Questo è particolarmente vero quando AWS Control Tower viene utilizzato.

Questo modello fornisce una soluzione prescrittiva che risolve efficacemente il problema utilizzando modifiche consolidate alla configurazione delle risorse e agendo su tali modifiche per generare risultati. La soluzione è progettata per scenari in cui sono presenti diversi CloudFormation stack creati in più di uno Regione AWS, in più di un account o in una combinazione di entrambi. Gli obiettivi della soluzione sono i seguenti:

- Semplifica il processo di rilevamento della deriva
- Imposta notifiche e avvisi
- Configura il reporting consolidato

Prerequisiti e limitazioni

Prerequisiti

- AWS Config abilitato in tutte le regioni e gli account che devono essere monitorati

Limitazioni

- Il report generato supporta solo i formati di output con valori separati da virgole (CSV) e JSON.

Architecture

Il diagramma seguente mostra AWS Organizations la configurazione con più account. AWS Config le regole comunicano tra gli account.

Il flusso di lavoro include i seguenti passaggi:

1. La AWS Config regola rileva la deriva.
2. I risultati del rilevamento della deriva trovati in altri account vengono inviati all'account di gestione.
3. La CloudWatch regola Amazon chiama una AWS Lambda funzione.
4. La funzione Lambda interroga la AWS Config regola per ottenere risultati aggregati.
5. La funzione Lambda notifica Amazon Simple Notification Service (Amazon SNS), che invia una notifica via e-mail della deriva.

Automazione e scalabilità

La soluzione qui presentata è scalabile sia per le regioni che per gli account aggiuntivi.

Tools (Strumenti)

Servizi AWS

- AWS Configfornisce una visualizzazione dettagliata della configurazione delle AWS risorse del tuo Account AWS. Questo include le relazioni tra le risorse e la maniera in cui sono state configurate in passato, in modo che tu possa vedere come le configurazioni e le relazioni cambiano nel corso del tempo.
- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue AWS risorse e delle applicazioni su cui esegui AWS in tempo reale.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.

Epiche

Automatizza il rilevamento della deriva per CloudFormation

Operazione	Description	Competenze richieste
Crea l'aggregatore.	1. Accedi a Console di gestione AWS e apri la	Architetto del cloud

Operazione	Description	Competenze richieste
	AWS Config console in https://console.aws.amazon.com/config. 2. Crea un aggregatore nell'account di gestione. 3. Assicurati che la replica dei dati sia attivata in modo da AWS Config poter recuperare i dati dagli account di origine. 4. Seleziona tutte le regioni e gli account applicabili. Puoi selezionare gli account in base a AWS Organizations. Consigliamo questo approccio perché i nuovi account nell'organizzazione e fanno automaticamente parte dell'aggregatore.	
Crea una regola AWS gestita.	Aggiungi la regola `cloudformation-stack-drift-detection-check` AWS gestita. La regola richiede un valore di parametro `:cloudformationArn` . Inserisci il ruolo IAM Amazon Resource Name (ARN) che dispone delle autorizzazioni per rilevare la deriva dello stack. Il ruolo deve avere una politica di fiducia che AWS Config consenta di assumerlo.	Architetto del cloud

Operazione	Description	Competenze richieste
Crea la sezione di interroga zione avanzata dell'aggr egatore.	Per recuperare pile alla deriva da più fonti, crea la seguente query: `SELECT resourceId, configuration.driftInformation.stackDriftStatus WHERE resourceType = 'AWS::CloudFormation::Stack' AND configuration.driftInformation.stackDriftStatus IN ('DRIFTED')`	Architetto del cloud, sviluppatore
Automatizza l'esecuzione della query e la pubblicazione.	1. Crea una funzione Lambda utilizzando il codice allegato. Lambda pubblicherà i risultati in un argomento SNS fornito come variabile di ambiente nella funzione Lambda. 2. Per ricevere avvisi, crea un abbonamento e-mail all'argomento SNS.	Architetto del cloud, sviluppatore
Crea una CloudWatch regola.	Crea una CloudWatch regola basata sulla pianificazione per chiamare la funzione Lambda, responsabile degli avvisi.	Architetto del cloud

Risorse correlate

Risorse

- Che cos'è AWS Config?
- Aggregazione di dati multiaccount e più regioni
- Rilevamento delle modifiche di configurazione non gestite agli stack e alle risorse
- IAM: passa un ruolo IAM a uno specifico Servizio AWS
- Cos'è Amazon SNS?

Informazioni aggiuntive

Considerazioni

Consigliamo di utilizzare la soluzione presentata in questo schema anziché soluzioni personalizzate che prevedono chiamate API a intervalli specifici per avviare il rilevamento delle deviazioni su ogni CloudFormation stack o set di stack. Le soluzioni personalizzate che utilizzano chiamate API a intervalli specifici possono portare a un numero elevato di chiamate API e influire sulle prestazioni. A causa del numero di chiamate API, può verificarsi una limitazione. Un altro potenziale problema è il ritardo nel rilevamento se le modifiche alle risorse vengono identificate solo in base alla pianificazione.

Poiché i set di stack sono costituiti da pile, puoi utilizzare questa soluzione. Come parte della soluzione sono disponibili anche i dettagli delle istanze dello stack.

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Importa con successo un bucket S3 come stack AWS CloudFormation

Ram Kandaswamy, Amazon Web Services

Riepilogo

Se utilizzi risorse Amazon Web Services (AWS), come i bucket Amazon Simple Storage Service (Amazon S3), e desideri utilizzare un approccio infrastructure-as code (IaC), puoi importare le tue risorse in CloudFormation AWS e gestirle come uno stack.

Questo modello fornisce i passaggi per importare correttamente un bucket S3 come stack CloudFormation AWS. Utilizzando l'approccio di questo modello, puoi evitare possibili errori che potrebbero verificarsi se importi il bucket S3 con una singola azione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Una policy esistente per i bucket S3 e i bucket S3. Per ulteriori informazioni su questo argomento, consulta Quale policy sui bucket S3 devo usare per rispettare la regola AWS Config s3- bucket-ssl-requests-only nell'AWS Knowledge Center.
- Una chiave AWS Key Management Service (AWS KMS) esistente e il relativo alias. Per ulteriori informazioni su questo argomento, consulta Lavorare con gli alias nella documentazione di AWS KMS.
- Il CloudFormation modello `CloudFormation-template-S3-bucket` AWS di esempio (allegato), scaricato sul tuo computer locale.

Architecture

Il diagramma mostra il flusso di lavoro seguente:

1. L'utente crea un modello AWS in formato JSON o YAML. CloudFormation
2. Il modello crea uno CloudFormation stack AWS per importare il bucket S3.
3. Lo CloudFormation stack AWS gestisce il bucket S3 specificato nel modello.

Stack tecnologico

- AWS CloudFormation
- AWS Identity and Access Management (IAM)
- AWS KMS
- Simple Storage Service (Amazon S3)

Strumenti

- AWS CloudFormation: AWS ti CloudFormation aiuta a creare e fornire distribuzioni di infrastrutture AWS in modo prevedibile e ripetuto.
- AWS Identity and Access Management (IAM): IAM è un servizio Web per controllare in modo sicuro l'accesso ai servizi AWS.
- AWS KMS — AWS Key Management Service (AWS KMS) è un servizio di crittografia e gestione delle chiavi scalato per il cloud.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è uno storage per Internet.

Epiche

Importa un bucket S3 con crittografia AWS KMS key basata su uno stack AWS CloudFormation

Operazione	Description	Competenze richieste

Crea un modello per importare il bucket S3 e la chiave KMS.	Sul tuo computer locale, crea un modello per importare il bucket S3 e la chiave KMS utilizzando il seguente modello di esempio:
	`AWSTemplateFormatVersion: 2010-09-09`
	`Parameters:`
	`  bucketName:`
		AWS DevOps

Operazione	Description	Competenze richieste
	Type: String Resources: S3Bucket: Type: 'AWS::S3::Bucket' DeletionPolicy: Retain Properties: BucketName: !Ref bucketName BucketEncryption: ServerSideEncryptionConfiguration: - ServerSideEncryptionByDefault: SSEAlgorithm: 'aws:kms' KMSMasterKeyID: !GetAtt - KMSS3Encryption - Arn KMSS3Encryption: Type: 'AWS::KMS::Key'	

Operazione	Description	Competenze richieste

	DeletionPolicy: Retain
	
	Properties:
	
	Enabled: true
	
	KeyPolicy: !Sub |-
	
	{
	
	"Id": "key-consolepolicy-3",
	
	"Version": "2012-10-17",
	
	"Statement": [
	
	{
	
	"Sid": "Enable IAM User Permissions",
	
	"Effect": "Allow",
	
	"Principal": {
	
	"AWS": ["arn:aws:iam::${AWS::AccountId}:root"]
	
	},
	
	"Action": "kms:*",	

Operazione	Description	Competenze richieste
	"Resource": "*" } }] } EnableKey Rotation: true	
Creare lo stack.	1. Accedi alla Console di gestione AWS, apri la CloudFormation console AWS, scegli Visualizza stack, scegli Crea stack, quindi scegli Con risorse esistenti (importa risorse). 2. Scegli Carica un file modello, quindi carica il file modello che hai creato in precedenza. 3. Inserisci un nome per lo stack e configura le opzioni rimanenti in base alle tue esigenze. 4. Scegli Crea stack e attendi che lo stato dello stack cambi a. `IMPORT_COMPLETE`	AWS DevOps

Operazione	Description	Competenze richieste

Crea l'alias della chiave KMS.	1. Sulla CloudFormation console AWS, scegli Stacks, scegli il nome dello stack che hai creato in precedenza, scegli il riquadro Template, quindi scegli Visualizza in Designer.
	2. Aggiungi il seguente frammento alla `Resource` sezione del modello, quindi scegli Create stack e completa la procedura guidata:
	
	`KMSS3EncryptionAlias:`
	
	`Type: 'AWS::KMS ::Alias'`
	
	`DeletionPolicy: Retain`
	
	`Properties:`
	
	`AliasName: alias/ S3BucketKey`
	
	`TargetKeyId: !Ref KMSS3Encryption`
	
	Per ulteriori informazioni a riguardo, consulta AWS CloudFormation stack
		AWS DevOps

Operazione	Description	Competenze richieste
	updates nella CloudFormation documentazione AWS.	

Operazione	Description	Competenze richieste

Aggiorna lo stack per includere la policy sui bucket S3.	1. Sulla CloudFormation console AWS, scegli Stacks, scegli il nome dello stack che hai creato in precedenza, scegli il riquadro Template, quindi scegli Visualizza in Designer.
	2. Aggiungi il seguente frammento alla `Resource` sezione del modello, quindi scegli Create stack e completa la procedura guidata:
	
	`S3BucketPolicy:`
	
	`  Type: 'AWS::S3::BucketPolicy'`
	
	`  Properties:`
	
	`    Bucket: !Ref S3Bucket`
	
	`    PolicyDocument: !Sub |-`
	
	`      {`
	
	`  "Version": "2008-10-17",`
	
	`                  "Id": "restricthttp",`
		AWS DevOps

Operazione	Description	Competenze richieste

	"Statement": [
	{
	"Sid": "denyhttp",
	"Effect": "Deny",
	"Principal": {
	"AWS": "*"
	},
	"Action": "s3:*",
	"Resource": ["arn:aws:s3:::${S3Bucket}","arn:aws:s3:::${S3Bucket}/*"],
	"Condition": {
	"Bool": {
	"aws:SecureTransport": "false"
	}	

Operazione	Description	Competenze richieste
	} }] } **Note** Questa policy sui bucket S3 include una dichiarazione di negazione che limita le chiamate API che non sono sicure.	

Operazione	Description	Competenze richieste
Aggiorna la politica chiave.	1. Sulla CloudFormation console AWS, scegli Stacks, scegli il nome dello stack che hai creato in precedenza, scegli il riquadro Template, quindi scegli Visualizza in Designer. 2. Modifica la risorsa KMS del modello per includere la policy chiave che consente agli amministratori di amministrare la chiave KMS. 3. Scegli Crea stack, scegli Avanti, quindi completa la procedura guidata in base alle tue esigenze. Per ulteriori informazioni, consulta le politiche chiave AWS KMS nella documenta zione di AWS KMS.	Amministratore AWS

Operazione	Description	Competenze richieste

Aggiungi tag a livello di risorsa.	1. Sulla CloudFormation console AWS, scegli Stacks, scegli il nome dello stack che hai creato in precedenza, scegli il riquadro Template, quindi scegli Visualizza in Designer.
	2. Aggiungi il seguente frammento alla sezione `Properties` delle risorse Amazon S3 del modello, quindi scegli Crea stack e completa la procedura guidata:
	`Tags:`
	`  - Key: createdBy`
	`  Value: Cloudformation`
		AWS DevOps

Risorse correlate

- Inserimento delle risorse esistenti nella CloudFormation gestione di AWS
- AWS re:Invent 2017: approfondimento su CloudFormation AWS (video)

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Sincronizza i dati tra i file system Amazon EFS in diverse regioni AWS utilizzando AWS DataSync

Sarat Chandra Pothula e Aditya Ambati, Amazon Web Services

Riepilogo

Questa soluzione fornisce un framework robusto per una sincronizzazione dei dati efficiente e sicura tra istanze Amazon Elastic File System (Amazon EFS) in diverse regioni AWS. Questo approccio è scalabile e fornisce una replica dei dati controllata in più regioni. Questa soluzione può migliorare le strategie di disaster recovery e ridondanza dei dati.

Utilizzando l'AWS Cloud Development Kit (AWS CDK), questo modello utilizza un approccio infrastructure-as code (IaC) per distribuire le risorse della soluzione. L'applicazione AWS CDK distribuisce le risorse essenziali di AWS, DataSync Amazon EFS, Amazon Virtual Private Cloud (Amazon VPC) e Amazon Elastic Compute Cloud (Amazon). EC2 Questo IaC fornisce un processo di distribuzione ripetibile e controllato dalla versione, completamente allineato con le best practice di AWS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- AWS Command Line Interface (AWS CLI) versione 2.9.11 o successiva, installata e configurata
- AWS CDK versione 2.114.1 o successiva, installata e avviata
- NodeJS versione 20.8.0 o successiva, installata

Limitazioni

- La soluzione eredita limitazioni da DataSync Amazon EFS, come velocità di trasferimento dei dati, limiti di dimensione e disponibilità regionale. Per ulteriori informazioni, consulta le DataSync quote AWS e le quote Amazon EFS.
- Questa soluzione supporta solo Amazon EFS. DataSync supporta altri servizi AWS, come Amazon Simple Storage Service (Amazon S3) e FSx Amazon for Lustre. Tuttavia, questa soluzione richiede modifiche per sincronizzare i dati con questi altri servizi.

Architecture

Questa soluzione implementa i seguenti stack CDK AWS:

- Stack Amazon VPC: questo stack configura risorse di cloud privato virtuale (VPC), tra cui sottoreti, un gateway Internet e un gateway NAT nelle regioni AWS primarie e secondarie.
- Stack Amazon EFS: questo stack distribuisce i file system Amazon EFS nelle regioni primarie e secondarie e li collega alle rispettive. VPCs
- Amazon EC2 stack: questo stack avvia EC2 istanze nelle regioni primarie e secondarie. Queste istanze sono configurate per montare il file system Amazon EFS, che consente loro di accedere allo storage condiviso.
- DataSync location stack: questo stack utilizza un costrutto personalizzato chiamato `DataSyncLocationConstruct` a creare risorse di DataSync localizzazione nelle regioni primarie e secondarie. Queste risorse definiscono gli endpoint per la sincronizzazione dei dati.
- DataSync stack di attività: questo stack utilizza un costrutto personalizzato chiamato `DataSyncTaskConstruct` per creare un' DataSync attività nella regione principale. Questa attività è configurata per sincronizzare i dati tra le regioni primarie e secondarie utilizzando le posizioni di DataSync origine e di destinazione.

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software che aiuta a definire e fornire l'infrastruttura cloud AWS in codice.
- AWS DataSync è un servizio di trasferimento e scoperta di dati online che ti aiuta a spostare file o dati di oggetti da, verso e tra i servizi di storage AWS.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon Elastic File System (Amazon EFS) ti aiuta a creare e configurare file system condivisi nel cloud AWS.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito. Questa rete virtuale è simile a una rete tradizionale che gestiresti nel tuo data center, con i vantaggi dell'utilizzo dell'infrastruttura scalabile di AWS.

Repository di codice

Il codice per questo modello è disponibile nel repository GitHub Amazon EFS Cross-Region DataSync Project.

Best practice

Segui le best practice descritte nella sezione Best practice per l'utilizzo del CDK AWS TypeScript per creare progetti IaC.

Epiche

Distribuisci l'app AWS CDK

Operazione	Description	Competenze richieste
Clona il repository del progetto.	Immetti il seguente comando per clonare il repository Amazon EFS Cross-Region DataSync Project. `git clone https://github.com/aws-samples/aws-efs-crossregion-datasync.git`	AWS DevOps
Installa le dipendenze npm.	Inserire il seguente comando. `npm ci`	AWS DevOps
Scegli le regioni primarie e secondarie.	Nel repository clonato, accedi alla directory. `src/infra` Nel `Launcher.ts` file, aggiorna i valori and. `PRIMARY_AWS_REGION` `SECONDARY_AWS_REGION` Usa i codici regionali corrispondenti.	AWS DevOps

Operazione	Description	Competenze richieste
	`const primaryRegion = { account: account, region: '<PRIMARY_AWS_REGION>' }; const secondaryRegion = { account: account, region: '<SECONDARY_AWS_REGION>' };`	
Avvia l'ambiente.	Inserisci il seguente comando per avviare l'account AWS e la regione AWS che desideri utilizzare. `cdk bootstrap <aws_account>/<aws_region>` Per ulteriori informazioni, consulta Bootstrapping nella documentazione di AWS CDK.	AWS DevOps
Elenca gli stack CDK AWS.	Immetti il seguente comando per visualizzare un elenco degli stack CDK AWS nell'app. `cdk ls`	AWS DevOps
Sintetizza gli stack CDK AWS.	Inserisci il seguente comando per creare un CloudFormation modello AWS per ogni stack definito nell'app AWS CDK. `cdk synth`	AWS DevOps

Operazione	Description	Competenze richieste
Distribuisci l'app AWS CDK.	Inserisci il seguente comando per distribuire tutti gli stack nel tuo account AWS, senza richiedere l'approvazione manuale per eventuali modifiche. `cdk deploy --all --require-approval never`	AWS DevOps

Convalida la distribuzione

Operazione	Description	Competenze richieste
Accedi all' EC2 istanza nella regione principale.	1. Utilizzando Session Manager, una funzionalità di AWS Systems Manager, accedi all' EC2 istanza nella regione principale. Per istruzioni, consulta Connect alla tua istanza Linux con AWS Systems Manager Session Manager. 2. Cambia le directory nel percorso di montaggio di Amazon EFS. `cd /mnt/efs`	AWS DevOps
Crea un file temporaneo.	Immetti il seguente comando per creare un file temporaneo nel percorso di montaggio di Amazon EFS.	AWS DevOps

Operazione	Description	Competenze richieste

	`sudo dd if=/dev/zero \ of=tmptst.dat \ bs=1G \ seek=5 \ count=0`
	
	`ls -lrt tmptst.dat`	

Avvia l' DataSync attività.	Inserisci il seguente comando per replicare il file temporaneo dalla regione principale alla regione secondaria, dove si `<ARN-task>` trova l'Amazon Resource Name (ARN) della DataSync tua attività.
	
	`aws datasync start-task-execution \ --task-arn <ARN-task>`
	
	Il comando restituisce l'ARN dell'esecuzione dell'attività nel formato seguente.
	
	`arn:aws:datasync:<region>:<account-ID>:task/task-execution/<exec-ID>`
		AWS DevOps

Operazione	Description	Competenze richieste
Controlla lo stato del trasferimento dei dati.	Immettere il comando seguente per descrivere l'attività di DataSync esecuzione, `<ARN-task-execution>` dov'è l'ARN dell'esecuzione dell'attività. `aws datasync describe-task-execution \ --task-execution-arn <ARN-task-execution>` L' DataSync attività è completa quando `PrepareStatus` `TransferStatus` , e `VerifyStatus` tutte hanno il valore`SUCCESS`.	AWS DevOps
Accedi all' EC2 istanza nella regione secondaria.	1. Utilizzando Session Manager, una funzionalità di AWS Systems Manager, accedi all' EC2 istanza nella regione secondaria. Per istruzioni, consulta Connect alla tua istanza Linux con AWS Systems Manager Session Manager. 2. Cambia le directory nel percorso di montaggio di Amazon EFS. `cd /mnt/efs`	AWS DevOps

Operazione	Description	Competenze richieste
Convalida la replica.	Inserisci il seguente comando per verificare che il file temporaneo esista nel file system Amazon EFS. `ls -lrt tmptst.dat`	AWS DevOps

Risorse correlate

Documentazione AWS

- Riferimento all'API CDK AWS
- Configurazione dei DataSync trasferimenti AWS con Amazon EFS
- Risoluzione dei problemi relativi ai DataSync trasferimenti AWS

Altre risorse AWS

- AWS DataSync FAQs

AWS Infrastruttura di test utilizzando LocalStack e Terraform Tests

Ivan Girardi e Ioannis Kalyvas, Amazon Web Services

Riepilogo

Questo modello ti aiuta a testare localmente l'infrastruttura come codice (IaC) AWS in Terraform senza la necessità di fornire l'infrastruttura nel tuo ambiente. AWS Integra il framework Terraform Tests con. LocalStack Il contenitore LocalStack Docker fornisce un ambiente di sviluppo locale che emula vari. Servizi AWS Questo ti aiuta a testare e iterare le implementazioni dell'infrastruttura senza incorrere in costi di. Cloud AWS

Questa soluzione offre i seguenti vantaggi:

- Ottimizzazione dei costi: l'esecuzione di test LocalStack elimina la necessità di utilizzo Servizi AWS. In questo modo si evitano i costi associati alla creazione, al funzionamento e alla modifica di tali risorse. AWS
- Velocità ed efficienza: i test locali sono inoltre in genere più rapidi rispetto alla distribuzione delle risorse. AWS Questo ciclo di feedback rapido accelera lo sviluppo e il debug. Poiché LocalStack funziona localmente, puoi sviluppare e testare i tuoi file di configurazione Terraform senza una connessione Internet. È possibile eseguire il debug dei file di configurazione Terraform localmente e ricevere un feedback immediato, il che semplifica il processo di sviluppo.
- Coerenza e riproducibilità: LocalStack fornisce un ambiente coerente per i test. Questa coerenza aiuta a garantire che i test producano gli stessi risultati, indipendentemente dalle AWS modifiche esterne o dai problemi di rete.
- Isolamento: Testing with LocalStack impedisce di influire accidentalmente sulle AWS risorse attive o sugli ambienti di produzione. Questo isolamento consente di sperimentare e testare diverse configurazioni in tutta sicurezza.
- Automazione: l'integrazione con una pipeline di integrazione e distribuzione continua (CI/CD) consente di testare automaticamente i file di configurazione Terraform. La pipeline testa a fondo l'IaC prima dell'implementazione.
- Flessibilità: è possibile simulare diverse Regioni AWS configurazioni e servizi per adattarle meglio agli ambienti di produzione. Account AWS

Prerequisiti e limitazioni

Prerequisiti

- Installa Docker
- Abilita l'accesso al socket Docker predefinito (). `/var/run/docker.sock` Per ulteriori informazioni, consulta la documentazione relativa ad LocalStack .
- Installa Docker Compose
- Installa Terraform versione 1.6.0 o successiva
- Installa Terraform CLI
- Configura il provider Terraform AWS
- (Facoltativo) Installa e configura AWS Command Line Interface (AWS CLI). Per un esempio di come utilizzare AWS CLI with LocalStack, consulta il repository GitHub Test AWS infrastructure using LocalStack and Terraform Tests.

Limitazioni

- Questo modello fornisce esempi espliciti per testare le risorse di Amazon Simple Storage Service (Amazon S3) AWS Lambda e AWS Step Functions Amazon DynamoDB. Tuttavia, puoi estendere questa soluzione per includere risorse aggiuntive. AWS
- Questo modello fornisce istruzioni per eseguire Terraform Tests localmente, ma è possibile integrare i test in qualsiasi CI/CD pipeline.
- Questo modello fornisce istruzioni per l'uso dell'immagine LocalStack comunitaria. Se stai usando l'immagine LocalStack Pro, consulta la documentazione LocalStack Pro.
- LocalStack fornisce servizi di emulazione per diversi. AWS APIs Per un elenco completo, vedere AWS Service Feature Coverage. Alcune funzionalità avanzate potrebbero richiedere un abbonamento a LocalStack Pro.

Architecture

Il diagramma seguente mostra l'architettura di questa soluzione. I componenti principali sono un repository di codice sorgente, una CI/CD pipeline e un LocalStack contenitore Docker. Il LocalStack Docker Container ospita localmente quanto segue: Servizi AWS

- Un bucket Amazon S3 per l'archiviazione di file
- Amazon CloudWatch per il monitoraggio e la registrazione
- Una AWS Lambda funzione per l'esecuzione di codice serverless
- Una macchina a AWS Step Functions stati per orchestrare flussi di lavoro in più fasi

- Una tabella Amazon DynamoDB per l'archiviazione di dati NoSQL

Il diagramma mostra il flusso di lavoro seguente:

1. Si aggiunge e si esegue il commit di un file di configurazione Terraform nel repository del codice sorgente.
2. La CI/CD pipeline rileva le modifiche e avvia un processo di compilazione per l'analisi statica del codice Terraform. La pipeline crea ed esegue il contenitore Docker. LocalStack Quindi la pipeline avvia il processo di test.
3. La pipeline carica un oggetto in un bucket Amazon S3 ospitato nel contenitore Docker. LocalStack
4. Il caricamento dell'oggetto richiama una funzione. AWS Lambda
5. La funzione Lambda memorizza la notifica degli eventi di Amazon S3 in un registro. CloudWatch
6. La funzione Lambda avvia una macchina a AWS Step Functions stati.
7. La macchina a stati scrive il nome dell'oggetto Amazon S3 in una tabella DynamoDB.
8. Il processo di test nella CI/CD pipeline verifica che il nome dell'oggetto caricato corrisponda alla voce nella tabella DynamoDB. Verifica inoltre che il bucket S3 sia distribuito con il nome specificato e che la funzione sia stata implementata correttamente. AWS Lambda

Tools (Strumenti)

Servizi AWS

- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue AWS risorse e delle applicazioni su cui esegui AWS in tempo reale.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Step Functionsè un servizio di orchestrazione senza server che consente di combinare AWS Lambda funzioni e altro Servizi AWS per creare applicazioni aziendali critiche.

Altri strumenti

- Docker è un insieme di prodotti Platform as a Service (PaaS) che utilizzano la virtualizzazione a livello di sistema operativo per fornire software in container.
- Docker Compose è uno strumento per definire ed eseguire applicazioni multi-contenitore.
- LocalStackè un emulatore di servizi cloud che viene eseguito in un unico contenitore. Utilizzando LocalStack, è possibile eseguire carichi di lavoro sul computer locale che utilizzano Servizi AWS, senza connettersi a. Cloud AWS
- Terraform è uno strumento IaC HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.
- Terraform Tests ti aiuta a convalidare gli aggiornamenti della configurazione dei moduli Terraform attraverso test analoghi all'integrazione o al test unitario.

Archivio di codice

Il codice per questo modello è disponibile nell'AWS infrastruttura di GitHub test utilizzando LocalStack il repository Terraform Tests.

Best practice

- Questa soluzione verifica AWS l'infrastruttura specificata nei file di configurazione Terraform e non distribuisce tali risorse in. Cloud AWS Se desideri distribuire le risorse, segui il principio del privilegio minimo (documentazione IAM) e configura correttamente il backend Terraform (documentazione Terraform).
- Durante l'integrazione LocalStack in una CI/CD pipeline, ti consigliamo di non eseguire il contenitore Docker in modalità privilegio. LocalStack Per ulteriori informazioni, consulta Privilegi di runtime e funzionalità Linux (documentazione Docker) e Sicurezza per i runner autogestiti (documentazione). GitLab

Epiche

Implementazione della soluzione

Operazione	Description	Competenze richieste
Clonare il repository.	In una shell bash, inserisci il seguente comando. Questo	DevOps ingegnere

Operazione	Description	Competenze richieste
	clona l'AWS infrastruttura di test utilizzando LocalStack il repository Terraform Tests da: GitHub `git clone https://github.com/aws-samples/localstack-terraform-test.git`	

Esegui il LocalStack contenitore.	1. Inserisci il seguente comando per navigare nel repository clonato:
	`cd localstack-terraform-test`
	2. Inserisci il seguente comando per avviare il contenitore LocalStack Docker in modalità distaccata:
	`docker-compose up -d`
	3. Attendi che il contenitore LocalStack Docker sia operativo.
		DevOps ingegnere

Inizializza Terraform.	Inserisci il seguente comando per inizializzare Terraform: `terraform init`	DevOps ingegnere

Operazione	Description	Competenze richieste

Esegui test Terraform.	1. Inserisci il seguente comando per eseguire Terraform Tests:
	`terraform test`
	2. Verifica che tutti i test siano stati completati correttamente. L'output visualizzato dovrebbe essere simile al seguente:
	`Success! 3 passed, 0 failed.`
		DevOps ingegnere

Eliminare le risorse.	Inserisci il seguente comando per distruggere il LocalStack contenitore: `docker-compose down`	DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
`Error: reading DynamoDB Table Item (Files|README.md): empty` risultato durante l'esecuzione del `terraform test` comando.	1. Reinserire il `terraform test` comando. 2. Se ciò non risolve l'errore, modifica il file main.tf per aumentare il timeout di sonno a un valore superiore a 15 secondi: `resource "time_sleep" "wait" { create_duration = "15s" triggers = { s3_object = local.key_json }`

Problema	Soluzione
	}

Risorse correlate

- Guida introduttiva a Terraform: Guida per AWS CDK ed AWS CloudFormation esperti (Prescriptive Guidance)AWS
- Migliori pratiche per l'utilizzo di Terraform AWS Provider (Prescriptive Guidance)AWS
- Terraform CI/CD e continuazione dei test AWS con il nuovo Terraform Test Framework (post sul blog)AWS
- Accelerazione della distribuzione del software utilizzando LocalStack Cloud Emulator da Marketplace AWS (post sul blog)AWS

Informazioni aggiuntive

Integrazione con Actions GitHub

È possibile integrare LocalStack e Terraform Tests in una CI/CD pipeline utilizzando GitHub Actions. Per ulteriori informazioni, consulta la documentazione di GitHub Actions. Di seguito è riportato un esempio di file di configurazione di GitHub Actions:

```
name: LocalStack Terraform Test

on:
  push:
    branches:
      - '**'

  workflow_dispatch: {}

jobs:
  localstack-terraform-test:
    runs-on: ubuntu-latest

    steps:
    - uses: actions/checkout@v4

    - name: Build and Start LocalStack Container
```

```
      run: |
        docker compose up -d

    - name: Setup Terraform
      uses: hashicorp/setup-terraform@v3
      with:
        terraform_version: latest

    - name: Run Terraform Init and Validation
      run: |
        terraform init
        terraform validate
        terraform fmt --recursive --check
        terraform plan
        terraform show

    - name: Run Terraform Test
      run: |
        terraform test

    - name: Stop and Delete LocalStack Container
      if: always()
      run: docker compose down
```

Aggiornamento dei cluster SAP Pacemaker da a ENSA1 ENSA2

Gergely Cserdi e Balazs Sandor Skublics, Amazon Web Services

Riepilogo

Questo modello spiega i passaggi e le considerazioni per l'aggiornamento di un cluster SAP Pacemaker basato su Standalone Enqueue Server () a. ENSA1 ENSA2 Le informazioni contenute in questo modello si applicano ai sistemi operativi SUSE Linux Enterprise Server (SLES) e Red Hat Enterprise Linux (RHEL).

I cluster Pacemaker su SAP NetWeaver 7.52 o S/4HANA 1709 e versioni precedenti funzionano su un'architettura e sono configurati specificamente per. ENSA1 ENSA1 Se esegui i tuoi carichi di lavoro SAP su Amazon Web Services (AWS) e sei interessato a passare a questi ENSA2, potresti scoprire che la documentazione di SAP, SUSE e RHEL non fornisce informazioni complete. Questo modello descrive i passaggi tecnici necessari per riconfigurare i parametri SAP e i cluster Pacemaker da cui eseguire l'aggiornamento a. ENSA1 ENSA2 Fornisce esempi di sistemi SUSE, ma il concetto è lo stesso per i cluster RHEL.

> **Note**
>
> ENSA1 e ENSA2 sono concetti che riguardano solo le applicazioni SAP, quindi le informazioni contenute in questo modello non si applicano a SAP HANA o ad altri tipi di cluster.

> **Note**
>
> Tecnicamente, ENSA2 può essere utilizzato con o senza Enqueue Replicator 2. Tuttavia, l'alta disponibilità (HA) e l'automazione del failover (tramite una soluzione cluster) richiedono Enqueue Replicator 2. Questo modello utilizza il termine ENSA2 cluster per fare riferimento ai cluster con Standalone Enqueue Server 2 ed Enqueue Replicator 2.

Prerequisiti e limitazioni

Prerequisiti

- Un cluster funzionante ENSA1 che utilizza Pacemaker e Corosync su SLES o RHEL.

- Almeno due istanze Amazon Elastic Compute Cloud (Amazon EC2) in cui sono in esecuzione le istanze (ABAP) SAP Central Services (ASCS/SCS) ed Enqueue Replication Server (ERS).
- Conoscenza della gestione di applicazioni e cluster SAP.
- Accesso all'ambiente Linux come utente root.

Limitazioni

- ENSA1i cluster basati supportano solo un'architettura a due nodi.
- ENSA2i cluster basati non possono essere distribuiti su versioni NetWeaver SAP precedenti alla 7.52.
- EC2 le istanze nei cluster devono trovarsi in zone di disponibilità AWS diverse.

Versioni del prodotto

- SAP NetWeaver versione 7.52 o successiva
- A partire da S/4HANA 2020, sono supportati solo i cluster ENSA2
- Kernel 7.53 o successivo, che supporta Enqueue Replicator 2 ENSA2
- SLES per applicazioni SAP versione 12 o successiva
- RHEL per SAP con High Availability (HA) versione 7.9 o successiva

Architecture

Stack tecnologico di origine

- SAP NetWeaver 7.52 con SAP Kernel 7.53 o successivo
- Sistema operativo SLES o RHEL

Stack tecnologico Target

- SAP NetWeaver 7.52 con SAP Kernel 7.53 o successivo, incluso S/4HANA 2020 con piattaforma ABAP
- Sistema operativo SLES o RHEL

Architettura Target

Il diagramma seguente mostra una configurazione HA ASCS/SCS e istanze ERS basate su un ENSA2 cluster.

Confronto tra cluster ENSA1 e cluster ENSA2

SAP è stato introdotto ENSA2 come successore di. ENSA1 Un cluster ENSA1 basato sul sistema supporta un'architettura a due nodi in cui l' ASCS/SCS istanza esegue il failover su ERS quando si verifica un errore. Questa limitazione deriva dal modo in cui l' ASCS/SCS istanza recupera le informazioni della tabella di blocco dalla memoria condivisa del nodo ERS dopo il failover. ENSA2i cluster basati su Enqueue Replicator 2 eliminano questa limitazione, poiché l' ASCS/SCS istanza può raccogliere le informazioni di blocco dall'istanza ERS attraverso la rete. ENSA2i cluster basati possono avere più di due nodi, poiché non è più necessario eseguire il failover dell' ASCS/SCS istanza sul nodo ERS. (Tuttavia, in un ambiente ENSA2 cluster a due nodi, l' ASCS/SCS istanza continuerà a eseguire il failover sul nodo ERS perché non ci sono altri nodi nel cluster su cui eseguire il failover). ENSA2 è supportato a partire da SAP Kernel 7.50 con alcune limitazioni. Per una configurazione HA che supporta Enqueue Replicator 2, il requisito minimo è NetWeaver 7,52 (vedere SAP OSS Note 2630416). S/4HANA 1809 viene fornito con l' ENSA2 architettura consigliata per impostazione predefinita, mentre S/4HANA supporta solo a partire dalla versione 2020. ENSA2

Automazione e scalabilità

Il cluster HA nell'architettura di destinazione esegue automaticamente il failover ASCS su altri nodi.

Scenari per il passaggio a cluster ENSA2 basati

Esistono due scenari principali per l'aggiornamento ai ENSA2 cluster basati:

- Scenario 1: si sceglie di eseguire l'aggiornamento ENSA2 senza un aggiornamento SAP o una conversione S/4HANA, presupponendo che la versione SAP e la versione del kernel supportino. ENSA2
- Scenario 2: si passa a ENSA2 come parte di un aggiornamento o di una conversione (ad esempio, a S/4HANA 1809 o versione successiva) utilizzando SUM.

La sezione Epics illustra i passaggi per questi due scenari. Il primo scenario richiede la configurazione manuale dei parametri relativi a SAP prima di modificare la configurazione del cluster per. ENSA2 Nel secondo scenario, i file binari e i parametri relativi a SAP vengono distribuiti da SUM e l'unica attività rimanente è aggiornare la configurazione del cluster per HA. Si consiglia comunque

di convalidare i parametri SAP dopo aver utilizzato SUM. Nella maggior parte dei casi, la conversione S/4HANA è il motivo principale per l'aggiornamento del cluster.

Tools (Strumenti)

- Per i gestori di pacchetti del sistema operativo, consigliamo gli strumenti Zypper (per SLES) o YUM (per RHEL).
- Per la gestione dei cluster, consigliamo le shell crm (per SLES) o pcs (per RHEL).
- Strumenti di gestione delle istanze SAP come. SAPControl
- (Opzionale) Strumento SUM per l'aggiornamento della conversione S/4HANA.

Best practice

- Per le best practice per l'utilizzo dei carichi di lavoro SAP su AWS, consulta SAP Lens per AWS Well-Architected Framework.
- Considera il numero di nodi del cluster (pari o dispari) nella tua architettura multinodo. ENSA2
- Configura il ENSA2 cluster per SLES 15 in linea con lo standard di certificazione SAP S/4-HA-CLU 1.0.
- Salva o esegui sempre il backup dello stato esistente del cluster e dell'applicazione prima di eseguire l'aggiornamento a. ENSA2

Epiche

Configura manualmente i parametri SAP per ENSA2 (solo scenario 1)

Operazione	Description	Competenze richieste
Configura i parametri nel profilo predefinito.	Se desideri eseguire l'aggiorn amento ENSA2 mantenendo la stessa versione SAP o se la versione di destinazione è predefinita ENSA1, imposta i parametri nel profilo predefini to (file DEFAULT.PFL) sui seguenti valori.	SAP

Operazione	Description	Competenze richieste

	`enq/enable=TRUE`
	`enq/serverhost=sapascsvirt`
	`enq/serverinst=10`
	`(instance number of ASCS/SCS instance)`
	`enque/process_location=REMOTESA`
	`enq/replicatorhost=sapersvirt`
	`enq/replicatorinst=11`
	`(instance number of ERS instance)`
	
	dove `sapascsvirt` è il nome host virtuale per le istanze ASCS ed è il nome host virtuale per le istanze ERS. `sapersvirt` È possibile modificarli per adattarli all'ambiente di destinazione.
	
	Note
	Per utilizzare questa opzione di aggiornamento, la versione SAP e la versione del kernel devono supportare Enqueue ENSA2 Replicator 2.	

Operazione	Description	Competenze richieste
Configura il profilo dell'istanza. ASCS/SCS	Se desideri eseguire l'aggiornamento ENSA2 mantenendo la stessa versione SAP o se la versione di destinazione è predefinita ENSA1, imposta i seguenti parametri nel profilo dell'istanza. ASCS/SCS La sezione del profilo in cui ENSA1 è definito ha un aspetto simile alla seguente. `#----------------- ----------------- ----------------- --------- Start SAP enqueue server #------------------ ----------------- ----------------- ------- _EN = en.sap$(S APSYSTEMNAME)$(INS TANCE_NAME) Execute_04 = local rm - f $(_EN) Execute_05 = local ln - s -f $(DIR_EXECUTABLE)/ enserver$(FT_EXE) $(_EN) Start_Program_01 = local $(_EN) pf=$(_PF)` Per riconfigurare questa sezione per ENSA2:	SAP

Operazione	Description	Competenze richieste
	1. Modifica il prefisso del `_EN` programma in `_ENQ` base alle informazioni più recenti di SAP (Nota OSS 2501860; richiede un account utente SAP ONE Support Launchpad). 2. Cambia il file binario per il server di enqueue da a. `enserver` `enq_server` 3. Imposta il nuovo parametro `suenq/server/replication/enable` . `TRUE` 4. Assicuratevi che`Autostart = 0`. Questa sezione del profilo avrebbe un aspetto simile alla seguente dopo le modifiche. `#------------------ ------------------ ------------------ --------- Start SAP enqueue server #------------------ ------------------ ------------------ ------- _ENQ = enq.sap$( SAPSYSTEMNAME)$(IN STANCE_NAME) Execute_04 = local rm - f $(_ENQ) Execute_05 = local ln - s -f $(DIR_EXECUTABLE)/`	

Operazione	Description	Competenze richieste

	`enq_server$(FT_EXE) $(_ENQ)`
	`Start_Program_01 = local $(_ENQ) pf= $(_PF)`
	`...`
	`enq/server/replication/enable = TRUE`
	`Autostart = 0`
	
	Important
	_ENQnon deve avere l'opzione di riavvio abilitata . Se `RestartProgram_01` è impostato per_ENQ, cambialo in`StartProgram_01` . Ciò impedisce a SAP di riavviare il servizio o di interferire con le risorse gestite dal cluster.	

Operazione	Description	Competenze richieste
Configura il profilo ERS.	Se desideri eseguire l'aggiornamento ENSA2 mantenendo la stessa versione SAP o se la versione di destinazione è predefinita ENSA1, imposta i seguenti parametri nel profilo dell'istanza ERS. Trova la sezione in cui è definito il replicatore di enqueue. Sarà simile al seguente. `#------------------ ------------------ ------------------- Start enqueue replicati on server #------------------- ------------------ ------------------ _ER = er.sap$(S APSYSTEMNAME)$(INS TANCE_NAME) Execute_03 = local rm - f $(_ER) Execute_04 = local ln - s -f $(DIR_EXECUTABLE)/ enrepserver$(FT_EXE) $(_ER) Start_Program_00 = local $(_ER) pf=$(_PF) NR=$(SCSID)` Per riconfigurare questa sezione per Enqueue Replicator 2:	SAP

Operazione	Description	Competenze richieste

	1. Modifica il prefisso del `_ER` programma in `_ENQR` base alle note più recenti di SAP (Nota OSS 2501860; richiede un account utente SAP ONE Support Launchpad).
	2. Cambia il file binario per il replicatore di enqueue in invece di. `enq_replicator enrepserver`
	3. Assicuratevi che. `Autostart = 0`
	
	Dopo le modifiche, questa sezione del profilo dovrebbe avere un aspetto simile alla seguente.
	
	`#------------------------------------------------------------`
	`Start enqueue replication server`
	`#------------------------------------------------------------`
	`_ENQR = enqr.sap$(SAPSYSTEMNAME)$(INSTANCE_NAME)`
	`Execute_01 = local rm -f $(_ENQR)`
	`Execute_02 = local ln -s -f $(DIR_EXECUTABLE)/enq_replicator$(FT_EXE) $(_ENQR)`	

Operazione	Description	Competenze richieste

	`Start_Program_00 = local $(_ENQR) pf= $(_PF) NR=$(SCSID)`
	`…`
	`Autostart = 0`
	
	Important
	_ENQRnon deve avere l'opzione di riavvio abilitata . Se `RestartProgram_01` è impostato per_ENQR, cambialo in`StartProgram_01` . Ciò impedisce a SAP di riavviare il servizio o di interferire con i servizi gestiti dal cluster.	

Operazione	Description	Competenze richieste

Riavviare SAP Start Services.	Dopo aver modificato i profili descritti in precedenza in questa epopea, riavvia SAP Start Services per entrambi ASCS/SCS e ERS.
	
	`sapcontrol -nr 10 -function RestartService SCT`
	
	`sapcontrol -nr 11 -function RestartService SCT`
	
	dove `SCT` si riferisce all'ID del sistema SAP e supponendo che 10 e 11 siano i numeri di istanza delle istanze ERS, ASCS/SCS rispettivamente.
		SAP

Riconfigurazione del cluster per ENSA2 (richiesto per entrambi gli scenari)

Operazione	Description	Competenze richieste
Verifica i numeri di versione nei Resource Agent SAP.	Quando si utilizza SUM per aggiornare SAP a S/4HANA 1809 o versioni successive, SUM gestisce le modifiche dei parametri nei profili SAP. Solo il cluster richiede una regolazione manuale. Tuttavia, si consiglia di verificare le impostazioni dei parametri prima di apportare modifiche al cluster.	Amministratore di sistema AWS

Operazione	Description	Competenze richieste

	Note
	Gli esempi di questo libro epico presuppongono che tu stia utilizzando il sistema operativo SUSE. Se utilizzi RHEL, dovrai usare strumenti come YUM e la shell pcs invece di Zypper e crm.
	
	Controlla entrambi i nodi dell'architettura per confermare che il `resource-agents` pacchetto corrisponda alla versione minima consigliata da SAP. Per SLES, controlla SAP OSS Note 2641019. Per RHEL, controlla SAP OSS Note 2641322. (SAP Notes richiede un account utente SAP ONE Support Launchpad).
	
	`sapers:sctadm 23>`
	`zypper search -s -i`
	`resource-agents`
	`Loading repository`
	`data...`
	`Reading installed`
	`packages...`	

Operazione	Description	Competenze richieste
	`S | Name | Type | Version | Arch | Repository --+------------ ----+---------+--- ---------- ---------- ---------

Operazione	Description	Competenze richieste

Controlla la configurazione del cluster.	Controlla la configurazione attuale del cluster.
	`crm configure show`
	Ecco un estratto dell'output completo:
	`node 1: sapascs`
	`node 2: sapers`
	`...`
	`primitive rsc_sap_S`
	`CT_ASCS10 SAPInstance`
	`\`
	`operations $id=rsc_s`
	`ap_SCT_ASCS10-oper`
	`ations \`
	`op monitor interval=120`
	`timeout=60 on-fail=r`
	`estart \`
	`params InstanceN`
	`ame=SCT_ASCS10_sap`
	`ascsvirt START_PRO`
	`FILE="/sapmnt/SCT/`
	`profile/SCT_ASCS10`
	`_sapascsvirt" \`
	`AUTOMATIC_RECOVER=`
	`false \`
	`meta resource-stickines`
	`s=5000 failure-t`
	`imeout=60 migration-`
	`threshold=1 priority=`
	`10`
	`primitive rsc_sap_S`
	`CT_ERS11 SAPInstance \`
	`operations $id=rsc_s`
	`ap_SCT_ERS11-opera`
	`tions \`
	`op monitor interval=120`
	`timeout=60 on-fail=r`
	`estart \`
		Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	`params InstanceN ame=SCT_ERS11_sape rsvirt START_PRO FILE="/sapmnt/SCT/ profile/SCT_ERS11_ sapersvirt" \ AUTOMATIC_RECOVER= false IS_ERS=true \ meta priority=1000 ... colocation col_sap_S CT_no_both -5000: grp_SCT_ERS11 grp_SCT_ASCS10 location loc_sap_S CT_failover_to_ers rsc_sap_SCT_ASCS10 \ rule 2000: runs_ers_SCT eq 1 order ord_sap_S CT_first_start_asc s Optional: rsc_sap_S CT_ASCS10:start rsc_sap_SCT_ERS11: stop symmetrical=false ...`  dove `sapascsvirt` si riferisce al nome host virtuale per le istanze ASCS, `sapersvirt` si riferisce al nome host virtuale per le istanze ERS e SCT si riferisce all'ID del sistema SAP.	

Operazione	Description	Competenze richieste
Rimuovi il vincolo di colocation del failover.	Nell'esempio precedente, il vincolo di posizione `loc_sap_SCT_failover_to_ers` specifica che la ENSA1 funzionalità di ASCS deve sempre seguire l'istanza ERS in caso di failover. Con ENSA2, ASCS dovrebbe essere in grado di eseguire il failover liberamente su tutti i nodi partecipanti, in modo da poter rimuovere questo vincolo. `crm configure delete loc_sap_SCT_failover_to_ers`	Amministratore di sistema AWS

Operazione	Description	Competenze richieste

Modifica le primitive.	Dovrai anche apportare piccole modifiche alle primitive ASCS ed ERS. SAPInstance
	
	Ecco un esempio di SAPInstance primitiva ASCS configurata per. ENSA1
	
	`primitive rsc_sap_SCT_ASCS10 SAPInstance \`
	`operations $id=rsc_sap_SCT_ASCS10-operations \`
	`op monitor interval=120 timeout=60 on-fail=restart \`
	`params InstanceName=SCT_ASCS10_sapascsvirt START_PROFILE="/sapmnt/SCT/profile/SCT_ASCS10_sapascsvirt" \`
	`AUTOMATIC_RECOVER=false \`
	`meta resource-stickiness=5000 failure-timeout=60 migration-threshold=1 priority=10`
	
	Per eseguire l'aggiornamento a ENSA2, modifica questa configurazione nel modo seguente.
	
	`primitive rsc_sap_SCT_ASCS10 SAPInstance \`
		Amministratore di sistema AWS

Operazione	Description	Competenze richieste

	`operations $id=rsc_sap_SCT_ASCS10-operations \`
	`op monitor interval=120 timeout=60 on-fail=restart \`
	`params InstanceName=SCT_ASCS10_sapascsvirt START_PROFILE="/sapmnt/SCT/profile/SCT_ASCS10_sapascsvirt" \`
	`AUTOMATIC_RECOVER=false \`
	`meta resource-stickiness=3000`
	
	Questo è un esempio di SAPInstance primitiva ERS configurata per ENSA1.
	
	`primitive rsc_sap_SCT_ERS11 SAPInstance \`
	`operations $id=rsc_sap_SCT_ERS11-operations \`
	`op monitor interval=120 timeout=60 on-fail=restart \`
	`params InstanceName=SCT_ERS11_sapersvirt START_PROFILE="/sapmnt/SCT/profile/SCT_ERS11_sapersvirt" \`
	`AUTOMATIC_RECOVER=false IS_ERS=true \`
	`meta priority=1000`	

Operazione	Description	Competenze richieste

	Per eseguire l'aggiornamento a ENSA2, modifica questa configurazione nel modo seguente.
	
	`primitive rsc_sap_SCT_ERS11 SAPInstance \ operations $id=rsc_sap_SCT_ERS11-operations \ op monitor interval=120 timeout=60 on-fail=restart \ params InstanceName=SCT_ERS11_sapersvirt START_PROFILE="/sapmnt/SCT/profile/SCT_ERS11_sapersvirt" \ AUTOMATIC_RECOVER=false IS_ERS=true`
	
	È possibile modificare le primitive in vari modi. Ad esempio, è possibile modificar le in un editor come vi, come nell'esempio seguente.
	
	`crm configure edit rsc_sap_SCT_ERS11`	

Operazione	Description	Competenze richieste
Disabilita la modalità di manutenzione.	Disabilita la modalità di manutenzione sul cluster. `crm configure property maintenance-mode=" false"` Quando il cluster non è in modalità di manutenzione, tenta di portare online le istanze ASCS ed ERS con le nuove ENSA2 impostazioni.	Amministratore di sistema AWS

(Facoltativo) Aggiungi nodi del cluster

Operazione	Description	Competenze richieste
Rivedi le migliori pratiche.	Prima di aggiungere altri nodi, assicurati di comprendere le migliori pratiche, ad esempio se utilizzare un numero pari o dispari di nodi.	Amministratore di sistema AWS
Aggiungi nodi.	L'aggiunta di altri nodi comporta una serie di attività, come l'aggiornamento del sistema operativo, l'install azione di pacchetti software corrispondenti ai nodi esistenti e la disponibilità dei mount. È possibile utilizzare l'opzione Prepare Additional Host in SAP Software Provision ing Manager (SWPM) per creare una baseline dell'host	Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	specifica per SAP. Per ulteriori informazioni, consulta le guide SAP elencate nella sezione successiva.	

Risorse correlate

Riferimenti SAP e SUSE

Per accedere a SAP Notes, è necessario disporre di un account utente SAP ONE Support Launchpad. Per ulteriori informazioni, vedere il sito Web del supporto SAP.

- SAP Note 2501860 – Documentazione per SAP Application Server per ABAP 7.52 NetWeaver
- SAP Note 2641019 – Installazione e aggiornamento da a in ambiente SUSE HA ENSA2 ENSA1 ENSA2
- SAP Note 2641322 – Installazione ENSA2 e aggiornamento da a ENSA2 quando si utilizzano le soluzioni Red Hat HA ENSA1 per SAP
- SAP Note 2711036 – Utilizzo dello standalone Enqueue Server 2 in un ambiente HA
- Enqueue Server 2 standalone (documentazione SAP)
- SAP S/4 HANA – Enqueue Replication 2 High Availability Cluster - Guida all'installazione (documentazione SUSE)

Riferimenti AWS

- SAP HANA su AWS: guida alla configurazione ad alta disponibilità per SLES e RHEL
- Lente SAP - AWS Well-Architected Framework

Usa zone di disponibilità coerenti in VPCs diversi account AWS

Adam Spicer, Amazon Web Services

Riepilogo

Sul cloud Amazon Web Services (AWS), una zona di disponibilità ha un nome che può variare tra i tuoi account AWS e un ID della zona di disponibilità (ID AZ) che ne identifica la posizione. Se utilizzi AWS CloudFormation per creare cloud privati virtuali (VPCs), devi specificare il nome o l'ID della zona di disponibilità durante la creazione delle sottoreti. Se VPCs crei più account, il nome della zona di disponibilità viene randomizzato, il che significa che le sottoreti utilizzano zone di disponibilità diverse in ciascun account.

Per utilizzare la stessa zona di disponibilità in tutti gli account, è necessario mappare il nome della zona di disponibilità di ciascun account allo stesso ID AZ. Ad esempio, il diagramma seguente mostra che l'ID `use1-az6` AZ è denominato `us-east-1a` nell'account AWS A e `us-east-1c` nell'account AWS Z.

Questo modello aiuta a garantire la coerenza zonale fornendo una soluzione scalabile e multiaccount per l'utilizzo delle stesse zone di disponibilità nelle sottoreti. La coerenza zonale assicura che il traffico di rete tra account eviti i percorsi di rete tra zone di disponibilità, il che aiuta a ridurre i costi di trasferimento dei dati e a ridurre la latenza di rete tra i carichi di lavoro.

Questo modello è un approccio alternativo alla CloudFormation AvailabilityZoneId proprietà AWS.

Prerequisiti e limitazioni

Prerequisiti

- Almeno due account AWS attivi nella stessa regione AWS.
- Valuta quante zone di disponibilità sono necessarie per supportare i requisiti VPC nella regione.
- Identifica e registra l'ID AZ per ogni zona di disponibilità che devi supportare. Per ulteriori informazioni su questo argomento, consulta la zona di disponibilità IDs per le tue risorse AWS nella documentazione di AWS Resource Access Manager.
- Un elenco ordinato e separato da virgole della tua AZ. IDs Ad esempio, la prima zona di disponibilità dell'elenco è mappata come`az1`, la seconda zona di disponibilità è mappata come e questa struttura di mappatura continua fino a quando l'elenco `az2` separato da virgole non è completamente mappato. Non esiste un numero massimo di AZ che è possibile mappare. IDs

- Il `az-mapping.yaml` file dal repository di mappatura GitHub Multi-account Availability Zone, copiato sul computer locale

Architecture

Il diagramma seguente mostra l'architettura che viene distribuita in un account e che crea i valori di AWS Systems Manager Parameter Store. Questi valori di Parameter Store vengono utilizzati quando si crea un VPC nell'account.

Il diagramma mostra il flusso di lavoro seguente:

1. La soluzione di questo modello viene implementata su tutti gli account che richiedono la coerenza zonale per un VPC.
2. La soluzione crea valori di Parameter Store per ogni ID AZ e memorizza il nuovo nome della zona di disponibilità.
3. Il CloudFormation modello AWS utilizza il nome della zona di disponibilità memorizzato in ogni valore di Parameter Store e ciò garantisce la coerenza zonale.

Il diagramma seguente mostra il flusso di lavoro per la creazione di un VPC con la soluzione di questo pattern.

Il diagramma mostra il flusso di lavoro seguente:

1. Invia un modello per creare un VPC ad AWS. CloudFormation
2. AWS CloudFormation risolve i valori del Parameter Store per ogni zona di disponibilità e restituisce il nome della zona di disponibilità per ogni ID AZ.
3. Viene creato un VPC con l'AZ corretta IDs richiesta per la coerenza zonale.

Dopo aver distribuito la soluzione di questo pattern, è possibile creare sottoreti che fanno riferimento ai valori del Parameter Store. Se usi AWS CloudFormation, puoi fare riferimento ai valori dei parametri di mappatura della zona di disponibilità dal seguente codice di esempio in formato YAML:

```
Resources:
    PrivateSubnet1AZ1:
```

```
        Type: AWS::EC2::Subnet
        Properties:
            VpcId: !Ref VPC
            CidrBlock: !Ref PrivateSubnetAZ1CIDR
            AavailabilityZone:
                !Join
                    - ''
                    - - '{{resolve:ssm:/az-mapping/az1:1}}'
```

Questo codice di esempio è contenuto nel `vpc-example.yaml` file del repository di mappatura GitHub Multi-account Availability Zone. Mostra come creare un VPC e delle sottoreti allineati ai valori del Parameter Store per garantire la coerenza zonale.

Stack tecnologico

- AWS CloudFormation
- AWS Lambda
- AWS Systems Manager Parameter Store

Automazione e scalabilità

Puoi implementare questo modello su tutti i tuoi account AWS utilizzando AWS CloudFormation StackSets o la soluzione Customizations for AWS Control Tower. Per ulteriori informazioni, consulta Working with AWS CloudFormation StackSets nella documentazione di AWS Cloudformation e Customizations for AWS Control Tower nella libreria di soluzioni AWS.

Dopo aver distribuito il CloudFormation modello AWS, puoi aggiornarlo per utilizzare i valori di Parameter Store e distribuirlo VPCs in pipeline o in base alle tue esigenze.

Tools (Strumenti)

Servizi AWS

- AWS ti CloudFormation aiuta a modellare e configurare le tue risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle per tutto il loro ciclo di vita. Puoi utilizzare un modello per descrivere le tue risorse e le loro dipendenze e lanciarle e configurarle insieme come uno stack, invece di gestire le risorse singolarmente. Puoi gestire e fornire stack su più account AWS e regioni AWS.
- AWS Lambda è un servizio di elaborazione che supporta l'esecuzione di codice senza effettuare il provisioning o la gestione di server. Lambda esegue il codice solo quando è necessario e si

dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo. Verrà addebitato soltanto il tempo di calcolo consumato e non verrà addebitato alcun costo quando il codice non è in esecuzione.

- AWS Systems Manager Parameter Store è una funzionalità di AWS Systems Manager. Fornisce uno storage sicuro e gerarchico per la gestione dei dati di configurazione e la gestione dei segreti.

Codice

Il codice per questo pattern è fornito nell'archivio di mappatura GitHub Multi-account Availability Zone.

Epiche

Distribuisci il file az-mapping.yaml

Operazione	Description	Competenze richieste
Determina le zone di disponibilità richieste per la regione.	1. Determina la zona di zona IDs che deve essere utilizzata in modo coerente nella tua regione. 2. Registra queste AZ IDs in un elenco separato da virgole e nell'ordine in cui desideri che vengano applicate. Ad esempio, la prima zona di disponibilità dell'elenco è mappata come `az1` e la seconda è mappata come. `az2` Non esiste un numero massimo di AZ IDs che è possibile mappare.	Architetto del cloud
Distribuisci il file az-mapping.yaml.	Usa il `az-mapping.yaml` file per creare uno CloudFormation stack AWS in tutti gli account AWS richiesti. Nel	Architetto del cloud

Operazione	Description	Competenze richieste
	`AZIds` parametro, usa l'elenco separato da virgole che hai creato in precedenza. Ti consigliamo di utilizzar e AWS CloudFormation StackSets o la soluzione Customizations for AWS Control Tower.	

Implementali VPCs nei tuoi account

Operazione	Description	Competenze richieste
Personalizza i CloudFormation modelli AWS.	Quando crei le sottoreti utilizzando AWS CloudForm ation, personalizza i modelli per utilizzare i valori di Parameter Store che hai creato in precedenza. Per un modello di esempio, consulta il `vpc-examp le.yaml` file nel repository di mappatura GitHub Multi-acc ount Availability Zone.	Architetto del cloud
Implementa il. VPCs	Implementa i CloudFormation modelli AWS personalizzati nei tuoi account. Ogni VPC nella regione ha quindi una coerenza zonale nelle zone di disponibilità utilizzate per le sottoreti.	Architetto del cloud

Risorse correlate

- Zona di disponibilità IDs per le tue risorse AWS (documentazione di AWS Resource Access Manager)
- AWS::EC2::Subnet(CloudFormation documentazione AWS)

Usa l'utente IDs nelle policy IAM per il controllo e l'automazione degli accessi

Srinivas Ananda Babu e Ram Kandaswamy, Amazon Web Services

Riepilogo

Questo modello spiega le potenziali insidie dell'utilizzo di politiche basate sul nome utente in AWS Identity and Access Management (IAM), i vantaggi dell'utilizzo degli utenti e come integrare questo approccio con l'automazione. IDs AWS CloudFormation

Nel Cloud AWS, il servizio IAM ti aiuta a gestire le identità degli utenti e il controllo degli accessi con precisione. Tuttavia, fare affidamento sui nomi utente nella creazione delle policy IAM può portare a rischi imprevisti per la sicurezza e problemi di controllo degli accessi. Ad esempio, considera questo scenario: un nuovo dipendente, John Doe, entra a far parte del tuo team e tu crei un account utente IAM con il nome utente`j.doe`, che concede loro le autorizzazioni tramite le policy IAM che fanno riferimento ai nomi utente. Quando John lascia l'azienda, l'account viene eliminato. Il problema inizia quando una nuova dipendente, Jane Doe, si unisce al team e il `j.doe` nome utente viene ricreato. Le politiche esistenti ora concedono a Jane Doe le stesse autorizzazioni di John Doe. Ciò crea un potenziale incubo in materia di sicurezza e conformità.

L'aggiornamento manuale di ogni policy in modo che rifletta i nuovi dettagli degli utenti è un processo lungo e soggetto a errori, soprattutto man mano che l'organizzazione cresce. La soluzione consiste nell'utilizzare un ID utente unico e immutabile. Quando crei un account utente IAM, AWS assegna all'utente IAM un ID utente univoco (o ID principale). Puoi utilizzare questi utenti IDs nelle tue policy IAM per garantire un controllo degli accessi coerente e affidabile, che non sia influenzato dalla modifica o dal riutilizzo del nome utente.

Ad esempio, una policy IAM che utilizza un ID utente potrebbe avere il seguente aspetto:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "s3:ListBucket",
            "Resource": "arn:aws:s3:::example-bucket",
            "Principal": { "AWS": "arn:aws:iam::123456789012:user/abcdef01234567890" }
        }
```

```
        ]
}
```

I vantaggi dell'utilizzo dell'utente IDs nelle politiche IAM includono:

- Unicità. IDs Gli utenti sono unici in tutto Account AWS, quindi forniscono una richiesta di autorizzazione corretta e coerente.
- Immutabilità. Gli utenti IDs non possono essere modificati, quindi forniscono un identificatore stabile per fare riferimento agli utenti nelle politiche.
- Controllo e conformità. Servizi AWS spesso includono l'utente IDs nei log e negli audit trail, il che semplifica la riconducibilità delle azioni a utenti specifici.
- Automazione e integrazione. L'utilizzo di script IDs di AWS APIs accesso SDKs utente o di automazione garantisce che i processi rimangano inalterati dalle modifiche al nome utente.
- A prova di futuro. L'utilizzo dell'utente IDs nelle politiche sin dall'inizio può prevenire potenziali problemi di controllo degli accessi o aggiornamenti estesi delle politiche.

Automazione

Quando si utilizzano strumenti di infrastruttura come codice (IaC), ad esempio AWS CloudFormation, le insidie delle policy IAM basate sui nomi utente possono ancora causare problemi. La risorsa utente IAM restituisce il nome utente quando si chiama la funzione intrinseca. `Ref` Man mano che l'infrastruttura dell'organizzazione si evolve, il ciclo di creazione ed eliminazione delle risorse, inclusi gli account utente IAM, può portare a problemi involontari di controllo degli accessi se si riutilizzano i nomi utente.

Per risolvere questo problema, ti consigliamo di incorporare l'utente nei tuoi modelli. IDs CloudFormation Tuttavia, ottenere un utente IDs per questo scopo può essere difficile. È qui che le risorse personalizzate possono essere utili. È possibile utilizzare risorse CloudFormation personalizzate per estendere le funzionalità del servizio mediante l'integrazione con AWS APIs i nostri servizi esterni. Creando una risorsa personalizzata che recupera l'ID utente di un determinato utente IAM, puoi rendere l'ID utente disponibile all'interno dei tuoi CloudFormation modelli. Questo approccio semplifica il processo di riferimento dell'utente IDs e garantisce che i flussi di lavoro di automazione rimangano solidi e a prova di futuro.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un ruolo IAM per un amministratore cloud per eseguire il CloudFormation modello

Limitazioni

- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Architecture

Architettura di Target

Il diagramma seguente mostra come CloudFormation utilizza una risorsa personalizzata supportata da AWS Lambda per recuperare l'ID utente IAM.

Automazione e scalabilità

Puoi utilizzare il CloudFormation modello più volte per diversi Regioni AWS account. È necessario eseguirlo solo una volta in ogni regione o account.

Tools (Strumenti)

Servizi AWS

- IAM — AWS Identity and Access Management (IAM) è un servizio web che ti aiuta a controllare in modo sicuro l'accesso alle AWS risorse. Utilizza IAM per controllare chi è autenticato (accesso effettuato) e autorizzato (dispone di autorizzazioni) per l'utilizzo di risorse.
- AWS CloudFormation— ti AWS CloudFormation aiuta a modellare e configurare AWS le tue risorse in modo da poter dedicare meno tempo alla gestione di tali risorse e più tempo a concentrarti sulle applicazioni che girano. AWS Crei un modello che descrive le AWS risorse che desideri e si CloudFormation occupa del provisioning e della configurazione di tali risorse per te.
- AWS Lambda— AWS Lambda è un servizio di elaborazione che supporta l'esecuzione di codice senza fornire o gestire server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo.

Best practice

Se stai partendo da zero o stai pianificando un'implementazione senza precedenti, ti consigliamo vivamente di utilizzarla AWS IAM Identity Centerper la gestione centralizzata degli utenti. IAM Identity Center si integra con i provider di identità esistenti (come Active Directory o Okta) per federare le identità degli utenti AWS, eliminando così la necessità di creare e gestire direttamente gli utenti IAM. Questo approccio non solo garantisce un controllo coerente degli accessi, ma semplifica anche la gestione del ciclo di vita degli utenti e aiuta a migliorare la sicurezza e la conformità in tutto l'ambiente. AWS

Epiche

Convalida le autorizzazioni

Operazione	Description	Competenze richieste
Convalida il tuo ruolo Account AWS e quello di IAM.	Conferma di avere un ruolo IAM con le autorizzazioni per distribuire CloudFormation modelli nel tuo. Account AWS Se hai intenzione di utilizzar e AWS CLI invece della CloudFormation console per distribuire il modello nell'ultimo passaggio di questa procedura , dovresti anche impostare credenziali temporanee per eseguire i comandi. AWS CLI Per istruzioni, consulta la documentazione IAM.	Architetto del cloud

Crea un CloudFormation modello

Operazione	Description	Competenze richieste
Crea un CloudFormation modello.	1. Crea un CloudFormation modello seguendo le	AWS DevOps, architetto del cloud

Operazione	Description	Competenze richieste
	istruzioni contenute nella CloudFormation documentazione. Puoi usare il formato JSON o YAML. Questo modello presuppone che tu stia utilizzando il formato YAML. 2. Salva il modello con il nome. `get_unique_user_id.yaml`	

Aggiungi un parametro di input per il nome utente.	Aggiungi il codice seguente alla `Parameters` sezione del CloudFormation modello:
	`Parameters:`
	`  NewIamUserName:`
	`  Type: String`
	`  Description: Unique username for the new IAM user`
	Questo parametro richiede all'utente il nome utente.
		AWS DevOps, architetto del cloud

Operazione	Description	Competenze richieste

Aggiungi una risorsa personalizzata per creare un utente IAM.	Aggiungi il codice seguente alla `Resources` sezione del CloudFormation modello:
	`Resources:`
	`  rNewIamUser:`
	`  Type: 'AWS::IAM ::User'`
	`  Properties:`
	`  UserName: !Ref NewIamUserName`
	Questo codice aggiunge una CloudFormation risorsa che crea un utente IAM con il nome fornito dal `NewIamUserName` parametro.
		AWS DevOps, architetto del cloud

Operazione	Description	Competenze richieste
Aggiungi un ruolo di esecuzione per la funzione Lambda.	In questo passaggio, crei un ruolo IAM che concede a una AWS Lambda funzione l'autorizzazione a ottenere l'IAM. `UserId` Specificate le seguenti autorizzazioni minime richieste per l'esecuzione di Lambda: • `logs:CreateLogStream` • `logs:PutLogEvents` • `CreateLogGroup` • `iam:GetUser` • `AssumeRole` per `lambda.amazonaws.com` Per istruzioni sulla creazione di un ruolo di esecuzione, consulta la documentazione Lambda. Farai riferimento a questo ruolo nel passaggio successivo, quando creerai la funzione Lambda.	Amministratore AWS, architetto cloud

Operazione	Description	Competenze richieste
Aggiungi una funzione Lambda per ottenere l'IAM unico. `UserId`	In questo passaggio, definisci una funzione Lambda con un runtime Python per ottenere l'IAM unico. `UserId` A tale scopo, aggiungete il codice seguente alla `Resources` sezione del CloudForm ation modello. Sostituiscilo `<<ROLENAME>>` con il nome del ruolo di esecuzion e che hai creato nell'ultimo passaggio. `GetUserLambdaFunct ion: Type: 'AWS::Lam bda::Function' Properties: Handler: index.han dler Role: <<ROLENAM E>> Timeout: 30 Runtime: python3.1 1 Code: ZipFile: | import cfnresponse, boto3 def handler(e vent, context): try: print(eve nt) user = boto3.client('iam' ).get_user(UserNam e=event['ResourceP`	AWS DevOps, architetto del cloud

Operazione	Description	Competenze richieste
	`roperties']['NewIa mUserName'])['User'] cfnrespon se.send(event, context, cfnrespon se.SUCCESS, {'NewIamU serId': user['Use rId'], 'NewIamUs erPath': user['Pat h'], 'NewIamUserArn': user['Arn']}) except Exception as e: cfnrespon se.send(event, context, cfnrespon se.FAILED, {'NewIamU ser': str(e)})`	
Aggiungi una risorsa personalizzata.	Aggiungi il codice seguente alla `Resources` sezione del CloudFormation modello: `rCustomGetUniqueUs erId: Type: 'Custom:: rCustomGetUniqueUs erIdWithLambda' Properties: ServiceToken: ! GetAtt GetUserLa mbdaFunction.Arn NewIamUserName: ! Ref NewIamUserName` Questa risorsa personalizzata chiama la funzione Lambda per ottenere l'IAM. `UserID`	AWS DevOps, architetto del cloud

Operazione	Description	Competenze richieste

Definisci CloudFormation gli output.	Aggiungi il codice seguente alla `Outputs` sezione del CloudFormation modello:
	`Outputs:`
	`  NewIamUserId:`
	`  Value: !GetAtt rCustomGetUniqueUserId.NewIamUserId`
	Questo mostra l'IAM `UserID` per il nuovo utente IAM.
		AWS DevOps, architetto del cloud

Salva il modello.	Salva le modifiche al CloudFormation modello.	AWS DevOps, architetto del cloud

Implementa il modello CloudFormation

Operazione	Description	Competenze richieste

Implementa il CloudFormation modello.	Per distribuire il `get_unique_user_id.yaml` modello utilizzando la CloudFormation console, segui le istruzioni nella CloudFormation documentazione.
	In alternativa, puoi eseguire il AWS CLI comando seguente per distribuire il modello:
	`aws cloudformation create-stack \`
	`--stack-name DemoNewUser \`
		AWS DevOps, architetto del cloud

Operazione	Description	Competenze richieste

	`--template-body file://get_unique_user_id.yaml \`
	`--parameters Parameter Key=NewIamUserName,ParameterValue=demouser \`
	`--capabilities CAPABILITY_NAMED_IAM`	

Risorse correlate

- Crea uno stack dalla CloudFormation console (CloudFormation documentazione)
- Risorse personalizzate supportate da Lambda (documentazione) CloudFormation
- Identificatori univoci (documentazione IAM)
- Utilizza credenziali temporanee con le AWS risorse (documentazione IAM)

Convalida il codice Account Factory for Terraform (AFT) localmente

Alexandru Pop e Michal Gorniak, Amazon Web Services

Riepilogo

Questo modello mostra come testare localmente il codice HashiCorp Terraform gestito da AWS Control Tower Account Factory for Terraform (AFT). Terraform è uno strumento Infrastructure as Code (IaC) che ti aiuta a utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud. AFT configura una pipeline Terraform che ti aiuta a fornire e personalizzare più in. Account AWS AWS Control Tower

Durante lo sviluppo del codice, può essere utile testare l'infrastruttura Terraform as code (IaC) localmente, al di fuori della pipeline AFT. Questo modello mostra come eseguire le seguenti operazioni:

- Recupera una copia locale del codice Terraform archiviato nei AWS CodeCommit repository del tuo account di gestione AFT.
- Simula la pipeline AFT localmente utilizzando il codice recuperato.

Questa procedura può essere utilizzata anche per eseguire comandi Terraform che non fanno parte della normale pipeline AFT. Ad esempio, è possibile utilizzare questo metodo per eseguire comandi come`terraform validate`, `terraform plan`e `terraform destroy`, e. `terraform import`

Prerequisiti e limitazioni

Prerequisiti

- Un ambiente AWS multi-account attivo che utilizza AWS Control Tower
- Un ambiente AFT completamente distribuito
- AWS Command Line Interface (AWS CLI), installato e configurato
- AWS CLI credenziali helper per AWS CodeCommit, installato e configurato
- Python 3.x
- Git, installato e configurato sul tuo computer locale
- `git-remote-commit`utilità, installata e configurata
- Terraform, installato e configurato (la versione locale del pacchetto Terraform deve corrispondere alla versione utilizzata nella distribuzione AFT)

Limitazioni

- Questo modello non copre le fasi di implementazione richieste per AWS Control Tower AFT o altri moduli Terraform specifici.
- L'output generato localmente durante questa procedura non viene salvato nei log di runtime della pipeline AFT.

Architecture

Stack tecnologico Target

- Infrastruttura AFT implementata all'interno di un'implementazione AWS Control Tower
- Terraform
- Git
- AWS CLI versione 2

Automazione e scalabilità

Questo modello mostra come richiamare localmente il codice Terraform per le personalizzazioni degli account globali AFT in un unico AFT gestito. Account AWS Dopo aver convalidato il codice Terraform, puoi applicarlo agli account rimanenti nel tuo ambiente multi-account. Per ulteriori informazioni, consulta Re-invoke customizations nella documentazione. AWS Control Tower

È inoltre possibile utilizzare un processo simile per eseguire personalizzazioni dell'account AFT in un terminale locale. Per richiamare localmente il codice Terraform dalle personalizzazioni dell'account AFT, clona il repository anziché il aft-account-customizationsrepository dal tuo account di gestione aft-global-account-customizationsAFT. CodeCommit

Tools (Strumenti)

Servizi AWS

- AWS Control Towerti aiuta a configurare e gestire un ambiente AWS multi-account, seguendo le migliori pratiche prescrittive.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando.

Altri servizi

- HashiCorp Terraform è uno strumento Infrastructure as Code (IaC) che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud.
- Git è un sistema di controllo delle versioni distribuito e open source.

Codice

Di seguito è riportato un esempio di script bash che può essere utilizzato per eseguire localmente il codice Terraform gestito da AFT. Per utilizzare lo script, segui le istruzioni nella sezione Epics di questo modello.

```
#! /bin/bash
# Version: 1.1 2022-06-24 Unsetting AWS_PROFILE since, when set, it interferes with
 script operation
#          1.0 2022-02-02 Initial Version
#
# Purpose: For use with AFT: This script runs the local copy of TF code as if it were
 running within AFT pipeline.
#        * Facilitates testing of what the AFT pipline will do
#           * Provides the ability to run terraform with custom arguments (like 'plan'
 or 'move') which are currently not supported within the pipeline.
#
# © 2021 Amazon Web Services, Inc. or its affiliates. All Rights Reserved.
# This AWS Content is provided subject to the terms of the AWS Customer Agreement
# available at http://aws.amazon.com/agreement or other written agreement between
# Customer and either Amazon Web Services, Inc. or Amazon Web Services EMEA SARL or
 both.
#
# Note: Arguments to this script are passed directly to 'terraform' without parsing nor
 validation by this script.
#
# Prerequisites:
#    1. local copy of ct GIT repositories
#    2. local backend.tf and aft-providers.tf filled with data for the target account
 on which terraform is to be run
#       Hint: The contents of above files can be obtain from the logs of a previous
 execution of the AFT pipeline for the target account.
#    3. 'terraform' binary is available in local PATH
#    4. Recommended: .gitignore file containing 'backend.tf', 'aft_providers.tf' so the
 local copy of these files are not pushed back to git
```

```
readonly credentials=$(aws sts assume-role \
    --role-arn arn:aws:iam::$(aws sts get-caller-identity --query "Account" --output
 text ):role/AWSAFTAdmin \
    --role-session-name AWSAFT-Session \
    --query Credentials )

unset AWS_PROFILE
export AWS_ACCESS_KEY_ID=$(echo $credentials | jq -r '.AccessKeyId')
export AWS_SECRET_ACCESS_KEY=$(echo $credentials | jq -r '.SecretAccessKey')
export AWS_SESSION_TOKEN=$(echo $credentials | jq -r '.SessionToken')
terraform "$@"
```

Epiche

Salva il codice di esempio come file locale

Operazione	Description	Competenze richieste
Salva il codice di esempio come file locale.	1. Copia lo script bash di esempio che si trova nella sezione Code di questo pattern e incollalo in un editor di codice. 2. Assegna un nome al file`ect_terraform.sh` , quindi salvalo localmente all'interno di una cartella dedicata, ad esempio `~/scripts` o`~/bin`.	Amministratore AWS
Rendi eseguibile il codice di esempio.	Apri una finestra di terminale ed esegui l'autenticazione nel tuo account di gestione AWS AFT effettuando una delle seguenti operazioni: • Utilizzate un AWS CLI profilo esistente configura to con le autorizzazioni	Amministratore AWS

Operazione	Description	Competenze richieste
	necessarie per accedere all'account di gestione AFT. Per utilizzare il profilo, puoi eseguire il seguente comando: `export AWS_PROFILE=<aft account profile name>` • Se la tua organizzazione utilizza SSO per accedere AWS, inserisci le credenziali del tuo account di gestione AFT nella pagina SSO dell'organizzazione. **Note** L'organizzazione potrebbe anche disporre di uno strumento personalizzato per fornire credenziali di autenticazione al proprio ambiente. AWS	

Operazione	Description	Competenze richieste

Verifica che l'accesso all'account di gestione AFT sia corretto Regione AWS.	**Important**
	Assicurati di utilizzare la stessa sessione di terminale con cui ti sei autenticato nel tuo account di gestione AFT.
	
	1. Passa alla tua distribuzione AFT Regione AWS eseguendo il seguente comando:
	`export AWS_REGION=<aft_region>`
	2. Assicurati di avere l'account corretto.
	a. Esegui il comando seguente:
	`aws code-commit list-repositories`
	b. Verifica che i repository elencati nell'output corrispondano ai nomi dei repository presenti nel tuo account di gestione AFT.
		Amministratore AWS

Operazione	Description	Competenze richieste

Crea una nuova directory locale per archiviare il codice del repository AFT.	Nella stessa sessione di terminale, esegui i seguenti comandi:
	`mkdir my_aft`
	`cd my_aft`
		Amministratore AWS

Operazione	Description	Competenze richieste

Clona il codice del repository AFT remoto.	1. Nel terminale locale, esegui il seguente comando:
	`git clone codecommit::$AWS_REGION://aft-global-customizations`
	Note
	Per semplicità, questa procedura e AFT utilizzano solo un ramo di codice principale. Per utilizzare la ramificazione del codice, puoi inserire anche i comandi di ramificazione del codice qui. Tuttavia, tutte le modifiche applicate dal ramo non principale verranno annullate quando l'automazione AFT applica il codice del ramo principale.
	2. Naviga nella directory clonata:
	`cd aft-global-customizations/terraform`
		Amministratore AWS

Crea i file di configurazione Terraform necessari per l'esecuzione locale della pipeline AFT

Operazione	Description	Competenze richieste
Apri una pipeline AFT precedentemente eseguita e copia i file di configurazione Terraform in una cartella locale.	**Note** I file di `aft-providers.tf` configurazione `backend.tf` e creati in questa epopea sono necessari per l'esecuzione locale della pipeline AFT. Questi file vengono creati automaticamente all'interno della pipeline AFT basata sul cloud, ma devono essere creati manualmente affinché la pipeline possa essere eseguita localmente. L'esecuzione locale della pipeline AFT richiede un set di file che rappresentino l'esecuzione della pipeline all'interno di un unico set. Account AWS 1. Utilizzando le credenziali AWS Control Tower del tuo account di gestione, accedi alla console e apri la	Amministratore AWS

Operazione	Description	Competenze richieste
	Console di gestione AWS stessa.AWS CodePipeline Assicurati di trovarti nello stesso Regione AWS posto in cui hai distribuito AFT. 2. Nel riquadro di navigazione a sinistra, seleziona Pipelines (Pipeline). 3. Scegliete ##########-customizations-pipeline. (Il ########## è l'ID che stai utilizzando per eseguire il codice Terraform localmente.) Account AWS 4. Assicurati che Most Recent Execution Marked mostri un valore Riuscito. Se il valore è diverso, è necessario richiamare nuovamente le personalizzazioni nella pipeline AFT. Per ulteriori informazioni, consulta Re-invoke customizations nella documentazione. AWS Control Tower 5. Scegli il runtime più recente per visualizzarne i dettagli. 6. Nella sezione Apply-AFT-Global-Customizations, trova lo stage Apply-Terraform. 7. Seleziona la sezione Dettagli dello stage Apply-Terraform.	

Operazione	Description	Competenze richieste

	8. Trova il log di runtime per la fase Apply-Terraform.
	9. Nel log di runtime, cerca la sezione che inizia e finisce con le seguenti righe:
	`"\n\n aft-provi ders.tf ... "\n \n backend.tf"`
	10. Copia l'output tra queste due etichette e salvale come file locale denominato `aft-providers.tf` all'interno della cartella Terraform locale (la directory di lavoro corrente della sessione terminale).
	Esempio di dichiarazione providers.tf generata automaticamente
	`## Autogenerated providers.tf ## ## Updated on: 2022-05-31 16:27:45 ## provider "aws" { region = "us-east-2" assume_role { role_arn = "arn:aws:iam::######## :role/AWSAFTExecution" }`	

Operazione	Description	Competenze richieste

	<pre> default_tags {
	    tags = {
	      managed_by
	             = "AFT"
	    }
	  }
	}</pre>
	11. Nel log di runtime, cercate la sezione che inizia e termina con le seguenti righe:
	<pre>"\n\n tf
	...
	"\n \n backend.tf"</pre>
	12. Copia l'output tra queste due etichette e salvale come file locale denominato `tf` all'interno della cartella Terraform locale (la directory di lavoro corrente della sessione terminale).
	
	Esempio di istruzione backend.tf generata automaticamente
	<pre>## Autogenerated
	 backend.tf ##
	## Updated on: 2022-05-3
	1 16:27:45 ##
	terraform {
	  required_version =
	 ">= 0.15.0"
	  backend "s3" {</pre>	

Operazione	Description	Competenze richieste

	<pre>region = "us-east-2"
	bucket = "aft-backend-###########-primary-region"
	key = "############-aft-global-customizations/terraform.tfstate"
	dynamodb_table = "aft-backend-#######"
	encrypt = "true"
	kms_key_id = "########-####-####-####-############"
	role_arn = "arn:aws:iam::#########:role/AWSAFTExecution"
	}
	}</pre>
	Note
	I `aft-providers.tf` file `backend.tf` and sono collegati a una distribuzione AFT e a una Account AWS cartella specifiche. Questi file sono inoltre diversi, a seconda che si trovino nel aft-account-custom	

Operazione	Description	Competenze richieste
	izationsrepository aft-global-customi zationsand all'interno della stessa distribuz ione AFT. Assicuratevi di generare entrambi i file dallo stesso elenco di runtime.	

Esegui la pipeline AFT localmente utilizzando lo script bash di esempio

Operazione	Description	Competenze richieste
Implementa le modifiche alla configurazione di Terraform che desideri convalidare.	1. Passa al aft-global-customi zationsrepository clonato eseguendo il seguente comando: `cd aft-global-customi zations/terraform` **Note** I file `backend.t f` e si `aft-provi ders.tf` trovano in questa directory . La directory contiene anche i file Terraform dal aft-global-customi zations repository.	Amministratore AWS

Operazione	Description	Competenze richieste
	2. Incorpora le modifiche al codice Terraform che desideri testare localmente nei file di configurazione.	

Operazione	Description	Competenze richieste

Esegui lo `ct_terraform.sh` script ed esamina l'output.	1. Accedere alla cartella locale che contiene lo script sh.
	2. Per convalidare il codice Terraform modificato, esegui lo `ct_terraform.sh` script eseguendo il seguente comando:
	`~/scripts/ct_terraform.sh apply`
	`terraform --help`
	Note
	Puoi eseguire qualsiasi comando Terraform durante questo passaggio. Per visualizzare un elenco completo dei comandi Terraform , esegui il seguente comando:
	3. Esamina l'output del comando, quindi esegui il debug delle modifiche al codice localmente prima di eseguire il commit e reinserire le modifiche nel repository AFT.
		Amministratore AWS

Operazione	Description	Competenze richieste
	Important • Tutte le modifiche apportate localmente e non trasferite all'archivio remoto sono temporanee e potrebbero essere annullate in qualsiasi momento da un'automazione della pipeline AFT in esecuzione. • L'automazione AFT può essere eseguita in qualsiasi momento, poiché può essere richiamata da altri utenti e dai trigger di automazione AFT. • AFT applicherà sempre il codice proveniente dal ramo principale del repository, annullando eventuali modifiche non eseguite.	

Invia le modifiche al codice locale nell'archivio AFT

Operazione	Description	Competenze richieste
Aggiungere riferimenti a `backend.tf` e `aft-providers.tf` file a un `.gitignore` file.	Aggiungi i `aft-providers.tf` file `backend.tf` and che hai creato a un `.gitignore` file eseguendo i seguenti comandi:	Amministratore AWS

Operazione	Description	Competenze richieste

	`echo backend.tf >> .gitignore`
	`echo aft-providers.tf >>.gitignore`
	
	Note
	Lo spostamento dei `.gitignore` file nel file garantisce che non vengano sottoposti a commit e reinseriti nell'archivio AFT remoto.	

Operazione	Description	Competenze richieste

Conferma e invia le modifiche al codice nell'archivio AFT remoto.	1. Per aggiungere nuovi file di configurazione Terraform al repository, esegui il seguente comando:
	`git add <filename>`
	2. Per confermare le modifiche e inviarle al repository AFT remoto CodeCommitt, esegui i seguenti comandi:
	`git commit -a`
	`git push`
	Important
	Le modifiche al codice introdotte seguendo questa procedura fino a questo punto vengono applicate a una Account AWS sola persona.
		Amministratore AWS

Implementa le modifiche a più account

Operazione	Description	Competenze richieste
Implementa le modifiche a tutti i tuoi account gestiti da AFT.	Per implementare le modifiche a più Account AWS personali zzazioni gestite da AFT, segui le istruzioni in Re-	Amministratore AWS

Operazione	Description	Competenze richieste
	invoke customizations nella documentazione. AWS Control Tower	

Altri modelli

- Aggiungi HA a Oracle PeopleSoft su Amazon RDS Custom utilizzando una replica di lettura
- Controlla automaticamente i gruppi AWS di sicurezza che consentono l'accesso da indirizzi IP pubblici
- Automatizza la creazione di account utilizzando Landing Zone Accelerator su AWS
- Automatizza l'aggiunta o l'aggiornamento delle voci di registro di Windows utilizzando AWS Systems Manager
- Automatizza la valutazione delle risorse AWS
- Automatizza il portafoglio e la distribuzione dei prodotti di AWS Service Catalog utilizzando AWS CDK
- Automatizza il failover e il failback tra regioni utilizzando DR Orchestrator Framework
- Eliminazione automatica degli AWS CloudFormation stack e delle risorse associate
- Automatizza l'inserimento e la visualizzazione di metriche personalizzate di Amazon MWAA su Amazon Managed Grafana utilizzando Terraform
- Automatizza la configurazione di RabbitMQ in Amazon MQ
- Automatizza l'implementazione dei Catena di approvvigionamento di AWS data lake in una configurazione multi-repository utilizzando GitHub Actions, Artifactory e Terraform
- Automatizza la replica delle istanze Amazon RDS su Account AWS
- Associa automaticamente una policy gestita da AWS per Systems Manager ai profili di EC2 istanza utilizzando Cloud Custodian e AWS CDK
- Crea automaticamente CI/CD pipeline e cluster Amazon ECS per microservizi utilizzando AWS CDK
- Rileva automaticamente le modifiche e avvia diverse CodePipeline pipeline per un monorepo in CodeCommit
- Crea una pipeline di dati per importare, trasformare e analizzare i dati di Google Analytics utilizzando il AWS DataOps Development Kit
- Create un Micro Focus Enterprise Server PAC con Amazon EC2 Auto Scaling e Systems Manager
- Crea e invia immagini Docker ad Amazon ECR utilizzando GitHub Actions e Terraform
- Crea una AWS landing zone che includa MongoDB Atlas
- Centralizza la gestione delle chiavi di accesso IAM in AWS Organizations utilizzando Terraform
- Centralizza la distribuzione dei pacchetti software in AWS Organizations utilizzando Terraform

- Configura la registrazione delle chiamate dei modelli in Amazon Bedrock utilizzando AWS CloudFormation
- Configurare il routing di sola lettura in un gruppo di disponibilità Always On in SQL Server su AWS
- Crea un portale per micro-frontend utilizzando Angular e AWS Amplify Module Federation
- Crea un framework di orchestrazione delle risorse basato su API utilizzando Actions e Terragrunt GitHub
- Crea una EventBridge connessione Amazon tra più account in un'organizzazione
- Crea automaticamente pipeline CI dinamiche per progetti Java e Python
- Implementa un'API Amazon API Gateway su un sito Web interno utilizzando endpoint privati e un Application Load Balancer
- Implementa e gestisci AWS Control Tower i controlli utilizzando AWS CDK e CloudFormation
- Implementa e gestisci AWS Control Tower i controlli utilizzando Terraform
- Implementa i canarini CloudWatch Synthetics utilizzando Terraform
- Implementa un cluster CockroachDB in Amazon EKS utilizzando Terraform
- Implementa un file system Lustre per l'elaborazione dei dati ad alte prestazioni utilizzando Terraform e DRA
- Implementa uno use case RAG AWS utilizzando Terraform e Amazon Bedrock
- Distribuisci risorse in una AWS Wavelength zona utilizzando Terraform
- Distribuisci istanze di cluster di failover di SQL Server su Amazon e EC2 Amazon FSx utilizzando Terraform
- Implementa le automazioni di sicurezza per la AWS WAF soluzione utilizzando Terraform
- Rileva le istanze di database Amazon RDS e Aurora con certificati CA in scadenza
- Documenta il progetto della tua landing zone AWS
- Esporta i report di AWS Backup da tutta l'organizzazione in AWS Organizations come file CSV
- Genera consigli personalizzati e riclassificati con Amazon Personalize
- Gestisci i set di autorizzazioni per più account utilizzando Account Factory for Terraform
- Identifica e avvisa quando le risorse Amazon Data Firehose non sono crittografate con una chiave AWS KMS
- Implementa Account Factory for Terraform (AFT) utilizzando una pipeline bootstrap
- Implementa il controllo delle versioni delle API basato sul percorso utilizzando domini personalizzati in Amazon API Gateway

- Installa SSM Agent sui nodi di lavoro Amazon EKS utilizzando Kubernetes DaemonSet
- Installa l'agente SSM e l' CloudWatch agente sui nodi di lavoro Amazon EKS utilizzando preBootstrapCommands
- Gestisci i set di AWS IAM Identity Center autorizzazioni come codice utilizzando AWS CodePipeline
- Gestisci i set di AWS autorizzazioni in modo dinamico utilizzando Terraform
- Gestisci i prodotti AWS Service Catalog in più account AWS e regioni AWS
- Gestisci le applicazioni container locali configurando Amazon ECS Anywhere con AWS CDK
- Gestisci AWS Organizations le policy come codice utilizzando AWS CodePipeline Amazon Bedrock
- Esegui la migrazione di record DNS in blocco verso una zona ospitata privata di Amazon Route 53
- Esegui la migrazione PeopleSoft da Oracle ad Amazon RDS Custom
- Migra i sistemi RHEL BYOL verso istanze con licenza AWS inclusa utilizzando AWS MGN
- Imposta uno spazio dati minimo praticabile per condividere i dati tra le organizzazioni
- Monitora ElastiCache i cluster Amazon per la crittografia a riposo
- Monitora l'attività delle applicazioni utilizzando CloudWatch Logs Insights
- Monitora i cluster SAP RHEL Pacemaker utilizzando i servizi AWS
- Crea un'architettura IPAM gerarchica e multiregionale utilizzando Terraform AWS
- Ottimizza le implementazioni serverless multi-account utilizzando i flussi di lavoro e Actions AWS CDK GitHub
- Esegui il provisioning di AWS Service Catalog prodotti basati su AWS CloudFormation modelli utilizzando GitHub Actions
- Fornisci ruoli IAM con privilegi minimi implementando una soluzione di distribuzione automatica di ruoli
- Rimuovi Amazon EC2 Entries Account AWS dall'altra AWS Managed Microsoft AD parte utilizzando l' AWS Lambda automazione
- Rimuovi EC2 le voci Amazon nello stesso Account AWS modulo AWS Managed Microsoft AD utilizzando l' AWS Lambda automazione
- Trasferimenti sicuri di file utilizzando Transfer Family, Amazon Cognito e GuardDuty
- Invia una notifica quando viene creato un utente IAM
- Configura un router cellulare serverless per un'architettura basata su celle
- Configura una CI/CD pipeline per carichi di lavoro ibridi su Amazon ECS Anywhere utilizzando AWS CDK e GitLab

- Configurare un' HA/DR architettura per Oracle E-Business Suite su Amazon RDS Custom con un database di standby attivo
- Configura la risoluzione DNS per reti ibride in un ambiente multi-account AWS
- Configura un'infrastruttura Multi-AZ per SQL Server Always On FCI utilizzando Amazon FSx
- Configura la funzionalità Oracle UTL_FILE su Aurora, compatibile con PostgreSQL
- Semplifica l'autenticazione delle applicazioni con TLS reciproco in Amazon ECS utilizzando Application Load Balancer
- Semplifica la gestione privata dei certificati utilizzando AWS Private CA e AWS RAM
- Semplifica i flussi di lavoro di machine learning dallo sviluppo locale agli esperimenti scalabili utilizzando SageMaker AI e Hydra
- Etichetta automaticamente gli allegati Transit Gateway utilizzando AWS Organizations
- Ruoli di transizione per un' PeopleSoft applicazione Oracle su Amazon RDS Custom for Oracle
- Usa Amazon Q Developer come assistente di programmazione per aumentare la tua produttività

App Web e mobili

Argomenti

- Autentica gli utenti delle applicazioni React esistenti utilizzando Amazon Cognito e l'interfaccia utente AWS Amplify
- Crea un'app React utilizzando AWS Amplify e aggiungi l'autenticazione con Amazon Cognito
- Crea un portale per micro-frontend utilizzando Angular e AWS Amplify Module Federation
- Implementa un'applicazione a pagina singola basata su React su Amazon S3 e CloudFront
- Implementa un'API Amazon API Gateway su un sito Web interno utilizzando endpoint privati e un Application Load Balancer
- Incorpora i componenti visivi di Amazon Quick Sight nelle applicazioni Web utilizzando Amazon Cognito e l'automazione IAc
- Esplora lo sviluppo completo di applicazioni web native per il cloud con Green Boost
- Esegui test unitari per un'applicazione Node.js GitHub utilizzando AWS CodeBuild
- Struttura un progetto Python in architettura esagonale utilizzando AWS Lambda
- Altri modelli

Autentica gli utenti delle applicazioni React esistenti utilizzando Amazon Cognito e l'interfaccia utente AWS Amplify

Daniel Kozhemyako, Amazon Web Services

Riepilogo

Questo modello mostra come aggiungere funzionalità di autenticazione a un'applicazione React frontend esistente utilizzando una libreria AWS Amplify UI e un pool di utenti Amazon Cognito.

Il modello utilizza Amazon Cognito per fornire l'autenticazione, l'autorizzazione e la gestione degli utenti per l'applicazione. Utilizza anche un componente di Amplify UI, una libreria open source che estende le funzionalità AWS Amplify di sviluppo dell'interfaccia utente (UI). Il componente Authenticator UI gestisce le sessioni di accesso ed esegue il flusso di lavoro connesso al cloud che autentica gli utenti tramite Amazon Cognito.

Dopo aver implementato questo modello, gli utenti possono accedere utilizzando una delle seguenti credenziali:

- Nome utente e password
- Provider di identità social, come Apple, Facebook, Google e Amazon
- Provider di identità aziendali compatibili con SAML 2.0 o con OpenID Connect (OIDC)

> **Note**
>
> Per creare un componente dell'interfaccia utente di autenticazione personalizzato, puoi eseguire il componente dell'interfaccia utente di Authenticator in modalità headless.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un'applicazione web React 18.2.0 o successiva
- Node.js e npm 6.14.4 o versioni successive, installati

Limitazioni

- Questo modello si applica solo alle applicazioni web React.
- Questo modello utilizza un componente Amplify UI predefinito. La soluzione non copre i passaggi necessari per implementare un componente dell'interfaccia utente personalizzato.

Versioni del prodotto

- Amplify UI 6.1.3 o versione successiva (Gen 1)
- Amplify 6.0.16 o versione successiva (Gen 1)

Architecture

Architettura Target

Il diagramma seguente mostra un'architettura che utilizza Amazon Cognito per autenticare gli utenti per un'applicazione web React.

Tools (Strumenti)

Servizi AWS

- Amazon Cognito fornisce autenticazione, autorizzazione e gestione degli utenti per app Web e mobili.

Altri strumenti

- Amplify UI è una libreria di interfaccia utente open source che fornisce componenti personalizzabili che puoi connettere al cloud.
- Node.js è un ambiente di JavaScript runtime basato sugli eventi progettato per la creazione di applicazioni di rete scalabili.
- npm è un registro software che viene eseguito in un ambiente Node.js e viene utilizzato per condividere o prendere in prestito pacchetti e gestire la distribuzione di pacchetti privati.

Best practice

Se stai creando una nuova applicazione, ti consigliamo di utilizzare Amplify Gen 2.

Epiche

Crea un pool di utenti Amazon Cognito

Operazione	Description	Competenze richieste
Crea un pool di utenti.	Crea un pool di utenti Amazon Cognito. Configura le opzioni di accesso e i requisiti di sicurezza del pool di utenti in base al tuo caso d'uso.	Sviluppatore di app
Aggiungi un client per l'app.	Configura un client di app per pool di utenti. Questo client è necessario per consentire all'applicazione di interagire con il pool di utenti di Amazon Cognito.	Sviluppatore di app

Integra il tuo pool di utenti Amazon Cognito con il componente Authenticator UI

Operazione	Description	Competenze richieste
Installare le dipendenze.	Per installare i `@aws-amplify/ui-react` pacchetti `aws-amplify` and, esegui il seguente comando dalla directory principale dell'applicazione: `npm i @aws-amplify/ui-react aws-amplify`	Sviluppatore di app
Configura il pool di utenti.	In base all'esempio seguente, create un `aws-exports.js` file e salvatelo nella `src`	Sviluppatore di app

Operazione	Description	Competenze richieste

	cartella. Il file deve includere le seguenti informazioni:
	• Regione AWS in cui si trova il tuo pool di utenti Amazon Cognito
	• ID del pool di utenti di Amazon Cognito
	• ID client dell'app
	
	`// replace the user pool region, id, and app client id details`
	`const awsmobile = { "aws_project_region": "put_your_region_here", "aws_cognito_region": "put_your_region_here", "aws_user_pools_id": "put_your_user_pool_id_here", "aws_user_pools_web_client_id": "put_your_user_pool_app_id_here" }`
	
	`export default awsmobile;`	

Operazione	Description	Competenze richieste

Importa e configura il servizio Amplify.	1. Nel file del punto di ingresso dell'applicazione (ad esempio,`App.js`), importate e caricate il `aws-exports.js` file inserendo le seguenti righe di codice:
	`import { Amplify } from 'aws-amplify'; import awsExports from './aws-exports';`
	2. In base all'esempio seguente, configura il client Amplify utilizzando il file: `aws-exports.js`
	`// Configure Amplify in index file or root file Amplify.configure({ ...awsExports });`
	Per ulteriori informazioni, consulta Configura le categorie Amplify nella documentazione di Amplify.
		Sviluppatore di app

Operazione	Description	Competenze richieste

Aggiungi il componente Authenticator UI.	Per visualizzare il componente dell'`Authenticator` interfaccia utente, aggiungete le seguenti righe di codice al file del punto di ingresso dell'applicazione (`App.js`):
	`import { Authenticator } from '@aws-amplify/ui-react';`
	`import '@aws-amplify/ui-react/styles.css';`
	Note
	Lo snippet di codice di esempio importa il componente `Authenticator` UI e il file Amplify UI styles.css, necessario quando si utilizzano i temi predefiniti del componente.
	Il componente `Authenticator` UI fornisce due valori restituiti:
	• Dettagli dell'utente
	• Una funzione che può essere invocata per disconnettere l'utente
		Sviluppatore di app

Operazione	Description	Competenze richieste
	Vedi il seguente componente di esempio: `function App() { return ( <Authenticator> {({ signOut, user }) => ( <div> <p>Welcome {user.username}</p> <button onClick={signOut}>Sign out</button> </div> )} </Authenticator> ); }` **Note** Per un `App.js` file di esempio, consultate la sezione Informazioni aggiuntive di questo modello.	

Operazione	Description	Competenze richieste

(Facoltativo) Recupera le informazioni sulla sessione.	Dopo l'autenticazione di un utente, puoi recuperare i dati relativi alla sua sessione dal client Amplify. Ad esempio, puoi recuperare il token web JSON (JWT) dalla sessione di un utente in modo da poter autenticare le richieste dalla sua sessione a un'API di backend.
	
	Guarda il seguente esempio di intestazione di richiesta che include un JWT:
	
	`import { fetchAuthSession } from 'aws-amplify/auth';`
	`(await fetchAuthSession()).tokens?.idToken?.toString();`
		Sviluppatore di app

risoluzione dei problemi

Problema	Soluzione
I nuovi utenti non possono iscriversi all'applicazione.	Come segue, assicurati che il tuo pool di utenti Amazon Cognito sia configurato per consentire agli utenti di registrarsi per far parte del pool di utenti: • Accedi a Console di gestione AWS, quindi apri la console Amazon Cognito. • Nel riquadro di navigazione a sinistra, scegli Pool di utenti.

Problema	Soluzione
	• Scegli il tuo pool di utenti dall'elenco. • In Impostazioni generali, scegli Politiche. • Scegli Consenti agli utenti di registrarsi.
Il componente di autenticazione ha smesso di funzionare dopo l'aggiornamento dalla v5 alla v6.	La `Auth` categoria è passata a un approccio funzionale e a parametri denominati in Amplify v6. Ora devi importare il funzionale APIs direttamente dal percorso. `aws-amplify/auth` Per ulteriori informazioni, consulta Migrare dalla v5 alla v6 nella documentazione di Amplify.

Risorse correlate

- Guida introduttiva ad Amazon Cognito (sito Web)AWS
- Crea una nuova app React (documentazione React)
- Cos'è Amazon Cognito? (documentazione di Amazon Cognito)
- Libreria Amplify UI (documentazione Amplify)

Informazioni aggiuntive

Il `App.js` file deve contenere il seguente codice:

```
import './App.css';
import { Amplify } from 'aws-amplify';
import awsExports from './aws-exports';
import { fetchAuthSession } from 'aws-amplify/auth';
import { Authenticator } from '@aws-amplify/ui-react';
import '@aws-amplify/ui-react/styles.css';
Amplify.configure({ ...awsExports });
let token = (await fetchAuthSession()).tokens?.idToken?.toString();
function App() {
  return (
      <Authenticator>
        {({ signOut, user }) => (
            <div>
```

```
              <p>Welcome {user.username}</p>
                <p>Your token is: {token}</p>
              <button onClick={signOut}>Sign out</button>
            </div>
        )}
      </Authenticator>
  );
}

export default App;
```

Crea un'app React utilizzando AWS Amplify e aggiungi l'autenticazione con Amazon Cognito

Rishi Singla, Amazon Web Services

Riepilogo

Questo modello dimostra come utilizzare AWS Amplify per creare un'app basata su React e come aggiungere l'autenticazione al frontend utilizzando Amazon Cognito. AWS Amplify è costituito da un set di strumenti (framework open source, ambiente di sviluppo visivo, console) e servizi (app Web e hosting di siti Web statici) per accelerare lo sviluppo di app mobili e Web su AWS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Node.js e npm installati sul tuo computer

Versioni del prodotto

- Node.js versione 10.x o successiva (per verificare la versione in uso, eseguilo `node -v` in una finestra di terminale)
- npm versione 6.x o successiva (per verificare la versione in uso, eseguila `npm -v` in una finestra di terminale)

Architecture

Stack tecnologico Target

- AWS Amplify
- Amazon Cognito

Tools (Strumenti)

- Interfaccia a riga di comando (CLI) Amplify
- Amplify Libraries (librerie client open source)

- Amplify Studio (interfaccia visiva)

Epiche

Installa AWS Amplify CLI

Operazione	Description	Competenze richieste

Installa la CLI Amplify.	L'Amplify CLI è una toolchain unificata per la creazione di servizi cloud AWS per la tua app React. Per installare la CLI Amplify, esegui:
	`npm install -g @aws-amplify/cli`
	npm ti avviserà se è disponibile una nuova versione principale. In tal caso, usa il seguente comando per aggiornare la tua versione di npm:
	`npm install -g npm@9.8.0`
	dove 9.8.0 si riferisce alla versione che si desidera installare.
		Sviluppatore di app

Crea un'app React

Operazione	Description	Competenze richieste

Crea un'app React.	Per creare una nuova app React, usa il comando:
	`npx create-react-app amplify-react-application`
	`ampify-react-application` dov'è il nome dell'app.
	Quando l'app è stata creata con successo, vedrai il messaggio:
	`Success! Created amplify-react-application`
	Verrà creata una directory con varie sottocartelle per l'app React.
		Sviluppatore di app

Avvia l'app sul tuo computer locale.	Vai alla directory `amplify-react-application` creata nel passaggio precedente ed esegui il comando: `amplify-react-application% npm start` Questo avvia l'app React sul tuo computer locale.	Sviluppatore di app

Configurazione della CLI Amplify

Operazione	Description	Competenze richieste
Configura Amplify per connetterti al tuo account AWS.	Configura Amplify eseguendo il comando: `amplify-react-application % amplify configure` La CLI di Amplify ti chiede di seguire questi passaggi per configurare l'accesso al tuo account AWS: 1. Accedi al tuo account amministratore AWS. 2. Specificare la regione AWS che si desidera utilizzare. 3. Crea un utente AWS Identity and Access Management (IAM) con accesso programmatico e allega la policy di `AdministratorAccess-Amplify` autorizzazione all'utente. 4. Crea e copia l'ID della chiave di accesso e la chiave di accesso segreta. 5. Inserisci questi dettagli nel terminale. 6. Crea un nome di profilo o usa il profilo predefinito.	General AWS, sviluppatore di app

Operazione	Description	Competenze richieste
	Warning Questo scenario richiede agli utenti IAM un accesso programmatico e credenziali a lungo termine, il che rappresenta un rischio per la sicurezza. Per ridurre questo rischio, si consiglia di fornire a questi utenti solo le autorizzazioni necessarie per eseguire l'attività e di rimuoverli quando non sono più necessari. Le chiavi di accesso possono essere aggiornate se necessario. Per ulteriori informazioni, consulta Updating access keys nella IAM User Guide. Questi passaggi vengono visualizzati nel terminale come segue. `Follow these steps to set up access to your AWS account:`	

Operazione	Description	Competenze richieste
	``` Sign in to your AWS administrator account: https://console.aw s.amazon.com/ Press Enter to continue Specify the AWS Region ? region: us-east-1 Follow the instructions at https://docs.am plify.aws/cli/start/ install/#configure- the-amplify-cli to complete the user creation in the AWS console https://console.aws .amazon.com/iamv2/ home#/users/create Press Enter to continue Enter the access key of the newly created user: ? accessKeyId: ******************** ? secretAccessKey: ****************** ****************** **** This would update/cr eate the AWS Profile in your local machine ? Profile Name: new Successfully set up the new user. ``` Per ulteriori informazioni su questi passaggi, consulta la documentazione nell'Amplify Dev Center.	

Inizializza Amplify

Operazione	Description	Competenze richieste

Inizializza Amplify.	1. Per inizializzare Amplify nella nuova directory, esegui:
	`amplify init`
	Amplify richiede il nome del progetto e i parametri di configurazione
	2. Specificate tutti i parametri , quindi premete Y per inizializzare il progetto con la configurazione specifica ta.
	`Project information`
	`| Name: amplifyre actproject`
	`| Environment: dev`
	`| Default editor: Visual Studio Code`
	`| App type: javascrip t`
	`| Javascript framework: react`
	`| Source Directory Path: src`
	`| Distribution Directory Path: build`
		Sviluppatore di app, General AWS

Operazione	Description	Competenze richieste

	<pre>| Build Command: npm
	 run-script build
	
	| Start Command: npm
	 run-script start</pre>
	3. Seleziona il profilo creato nel passaggio precedente. Le risorse verranno distribuite nell'`dev`ambiente del progetto Amplify che hai creato.
	4. Per confermare che le risorse sono state create, puoi aprire la console AWS Amplify e visualizzare il modello CloudFormation AWS utilizzato per creare le risorse e i dettagli.
	
	<pre>Deploying root stack
	 amplifyreactproject
	 [==================
	==----------------
	----] 2/4
	amplify-amplif
	yreactproject-d…
	 AWS::CloudFormatio
	n::Stack
	 CREATE_IN_PROGRESS
	
	UnauthRole
	 AWS::IAM:
	:Role
	 CREATE_COMPLETE</pre>	

Operazione	Description	Competenze richieste

	`DeploymentBucket AWS::S3::Bucket CREATE_IN_PROGRESS`
	
	`AuthRole AWS::IAM::Role CREATE_COMPLETE`	

Aggiungi l'autenticazione al frontend

Operazione	Description	Competenze richieste
Aggiungere l'autenticazione.	È possibile utilizzare il `amplify add <category>` comando per aggiungere funzionalità come l'accesso utente o un'API di backend. In questo passaggio utilizzerai il comando per aggiungere l'autenticazione. Amplify fornisce un servizio di autenticazione backend con Amazon Cognito, librerie frontend e un componente dell'interfaccia utente Authenticator drop-in. Le funzionalità includono la registrazione utente, l'accesso utente, l'autenticazione a più fattori, la disconnessione utente e l'accesso senza password. Puoi anche	Sviluppatore di app, General AWS

Operazione	Description	Competenze richieste

	autenticare gli utenti tramite l'integrazione con provider di identità federati come Amazon, Google e Facebook. La categoria di autenticazione Amplify si integra perfettam ente con altre categorie di Amplify come API, analisi e archiviazione, in modo da poter definire regole di autorizzazione per utenti autenticati e non autenticati.
	
	1. Per configurare l'autenti cazione per la tua app React, esegui il comando:
	
	`amplify-react-appl ication1 % amplify add auth`
	
	Questo mostra le seguenti informazioni e istruzion i. È possibile scegliere la configurazione appropriata in base ai requisiti aziendali e di sicurezza.
	
	`Using service: Cognito, provided by: awscloudformation The current configure d provider is Amazon Cognito. Do you want to use the default authentic ation and security`	

Operazione	Description	Competenze richieste
	configuration? (Use arrow keys) # Default configuration Default configuration with Social Provider (Federation) Manual configuration I want to learn more.	
	2. Per un semplice esempio, scegli la configurazione predefinita e quindi seleziona il meccanismo di accesso per gli utenti (in questo caso, e-mail): How do you want users to be able to sign in? Username # Email Phone Number Email or Phone Number I want to learn more.	

Operazione	Description	Competenze richieste

	3. Ignora le impostazioni avanzate per completare l'aggiunta di risorse di autenticazione:
	
	`Do you want to configure advanced settings? (Use arrow keys)`
	`# No, I am done.`
	
	`Yes, I want to make some additional changes.`
	
	4. Crea le tue risorse di backend locali ed esegui il provisioning nel cloud:
	
	`amplify-react-application1 % amplify push`
	
	Questo comando apporta le modifiche appropriate ai pool di utenti Congito nel tuo account.
	5. Premi Y per configurare la `auth` risorsa utilizzando CloudFormation.
	
	Questo configura le seguenti risorse:
	
	`UserPool AWS::Cognito::UserPool CREATE_COMPLETE`	

Operazione	Description	Competenze richieste

	`UserPoolClientWeb AWS::Cognito::UserPoolClient CREATE_COMPLETE`
	
	`UserPoolClientWeb AWS::Cognito::UserPoolClient CREATE_COMPLETE`
	
	`UserPoolClientRole AWS::IAM::Role CREATE_COMPLETE`
	
	`UserPoolClientLambda AWS::Lambda::Function CREATE_COMPLETE UserPoolClientLambdaPolicy AWS::IAM::Policy CREATE_COMPLETE`
	
	`UserPoolClientLogPolicy AWS::IAM::Policy CREATE_IN_PROGRESS`
	
	Puoi anche utilizzare la console AWS Cognito per visualizzare queste risorse (cerca i pool di utenti e i pool di identità di Cognito).
	
	Questo passaggio aggiorna il `aws-exports.js` file	

Operazione	Description	Competenze richieste
	nella `src` cartella dell'app React con le configurazioni del pool di utenti e del pool di identità di Cognito.	

Cambia il file App.js

Operazione	Description	Competenze richieste
Modificare il file App.js.	Nella `src` cartella, aprite e modificate il `App.js` file. Il file modificato dovrebbe avere il seguente aspetto: `{ App.Js File after modifications: import React from 'react'; import logo from './logo.svg'; import './App.css'; import { Amplify } from 'aws-amplify'; import { withAuthenticator, Button, Heading } from '@aws-amplify/ui-react'; import awsconfig from './aws-exports'; Amplify.configure(awsconfig); function App({ signOut }) { return (<div> <h1>Thankyou for doing verification</h1>`	Sviluppatore di app

Operazione	Description	Competenze richieste

	My Content
	<button onClick={signOut}>Sign out</button>
	
	);
	}
	export default withAuthenticator(App);	

Importa pacchetti React.	Il `App.js` file importa due pacchetti React. Installa questi pacchetti usando il comando: `amplify-react-application1 % npm install --save aws-amplify @aws-amplify/ui-react`	Sviluppatore di app

Avvia l'app React e controlla l'autenticazione

Operazione	Description	Competenze richieste
Avvia l'app.	Avvia l'app React sul tuo computer locale: `amplify-react-application1 % npm start`	Sviluppatore di app, General AWS
Verifica l'autenticazione.	Controlla se l'app richiede i parametri di autenticazione. (Nel nostro esempio, abbiamo configurato l'e-mail come metodo di accesso.)	Sviluppatore di app, General AWS

Operazione	Description	Competenze richieste
	L'interfaccia utente frontend dovrebbe richiedere le credenziali di accesso e fornire un'opzione per creare un account. Puoi anche configurare il processo di compilazione di Amplify per aggiungere il backend come parte di un flusso di lavoro di distribuz ione continua. Tuttavia, questo modello non copre questa opzione.	

Risorse correlate

- Guida introduttiva (documentazione npm)
- Crea un account AWS autonomo (documentazione AWS Account Management)
- Documentazione AWS Amplify
- Documentazione Amazon Cognito

Crea un portale per micro-frontend utilizzando Angular e AWS Amplify Module Federation

Milena Godau e Pedro Garcia, Amazon Web Services

Riepilogo

Un'architettura micro-frontend consente a più team di lavorare su diverse parti di un'applicazione frontend in modo indipendente. Ogni team può sviluppare, creare e distribuire un frammento del frontend senza interferire con altre parti dell'applicazione. Dal punto di vista dell'utente finale, sembra essere un'unica applicazione coesa. Tuttavia, interagiscono con diverse applicazioni indipendenti pubblicate da team diversi.

Questo documento descrive come creare un'architettura di micro-frontend utilizzando AWS Amplifyil framework di frontend Angular e Module Federation. In questo modello, i micro-frontend vengono combinati sul lato client da un'applicazione shell (o principale). L'applicazione shell funge da contenitore che recupera, visualizza e integra i micro-frontend. L'applicazione shell gestisce il routing globale, che carica diversi micro-frontend. Il plugin @angular -architects/module-federation integra Module Federation con Angular. Si distribuiscono l'applicazione shell e i micro-frontend utilizzando. AWS Amplify Gli utenti finali accedono all'applicazione tramite un portale basato sul Web.

Il portale è diviso verticalmente. Ciò significa che i micro-frontend sono intere viste o gruppi di visualizzazioni, anziché parti della stessa vista. Pertanto l'applicazione shell carica solo un micro-frontend alla volta.

I micro-frontend sono implementati come moduli remoti. L'applicazione shell carica pigramente questi moduli remoti, il che posticipa l'inizializzazione del microfrontend finché non è necessaria. Questo approccio ottimizza le prestazioni dell'applicazione caricando solo i moduli necessari. Ciò riduce il tempo di caricamento iniziale e migliora l'esperienza utente complessiva. Inoltre, condividi le dipendenze comuni tra i moduli tramite il file di configurazione webpack (webpack.config.js). Questa pratica promuove il riutilizzo del codice, riduce la duplicazione e semplifica il processo di raggruppamento.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS

- Node.js e npm, installati
- Amplify CLI, installata
- CLI angolare, installata
- Autorizzazioni da utilizzare AWS Amplify
- Familiarità con Angular

Versioni del prodotto

- Angular CLI versione 13.1.2 o successiva
- @angular -architects/module-federation versione 14.0.1 o successiva
- webpack versione 5.4.0 o successiva
- AWS Amplify Gen 1

Limitazioni

Un'architettura micro-frontend è un approccio potente per la creazione di applicazioni web scalabili e resilienti. Tuttavia, è fondamentale comprendere le seguenti potenziali sfide prima di adottare questo approccio:

- Integrazione: una delle sfide principali è il potenziale aumento della complessità rispetto ai frontend monolitici. Orchestrare più micro-frontend, gestire le comunicazioni tra di loro e gestire le dipendenze condivise può essere più complesso. Inoltre, potrebbe esserci un sovraccarico prestazionale associato alla comunicazione tra i micro-frontend. Questa comunicazione può aumentare la latenza e ridurre le prestazioni. Questo problema deve essere risolto attraverso meccanismi di messaggistica efficienti e strategie di condivisione dei dati.
- Duplicazione del codice: poiché ogni micro-frontend è sviluppato in modo indipendente, esiste il rischio di duplicare il codice per funzionalità comuni o librerie condivise. Ciò può aumentare le dimensioni complessive dell'applicazione e introdurre problemi di manutenzione.
- Coordinamento e gestione: coordinare i processi di sviluppo e implementazione su più micro-frontend può essere difficile. Garantire un controllo coerente delle versioni, gestire le dipendenze e mantenere la compatibilità tra i componenti diventa più importante in un'architettura distribuita. Stabilire una governance chiara, linee guida e pipeline automatizzate di test e implementazione è essenziale per una collaborazione e una distribuzione senza interruzioni.
- Test: testare le architetture di micro-frontend può essere più complesso rispetto al test di frontend monolitici. Richiede uno sforzo aggiuntivo e strategie di test specializzate per eseguire test e test

di integrazione tra componenti e per convalidare esperienze utente coerenti end-to-end su più microfrontend.

Prima di adottare l'approccio dei microfrontend, ti consigliamo di leggere Comprendere e implementare i microfrontend su. AWS

Architecture

In un'architettura di micro-frontend, ogni team sviluppa e implementa funzionalità in modo indipendente. L'immagine seguente mostra come più DevOps team collaborano. Il team del portale sviluppa l'applicazione shell. L'applicazione shell funge da contenitore. Recupera, visualizza e integra le applicazioni di micro-frontend pubblicate da altri team. DevOps Si utilizza AWS Amplify per pubblicare l'applicazione shell e le applicazioni micro-frontend.

Il diagramma dell'architettura mostra il seguente flusso di lavoro:

1. Il team del portale sviluppa e gestisce l'applicazione shell. L'applicazione shell orchestra l'integrazione e il rendering dei micro-frontend per comporre l'intero portale.
2. I team A e B sviluppano e gestiscono uno o più micro-frontend o funzionalità integrate nel portale. Ogni team può lavorare in modo indipendente sui rispettivi micro-frontend.
3. L'utente finale si autentica utilizzando Amazon Cognito.
4. L'utente finale accede al portale e l'applicazione shell viene caricata. Mentre l'utente naviga, l'applicazione shell si occupa del routing e recupera il micro-frontend richiesto, caricando il relativo pacchetto.

Tools (Strumenti)

Servizi AWS

- AWS Amplifyè un insieme di strumenti e funzionalità appositamente progettati che aiutano gli sviluppatori web e mobili di frontend a creare rapidamente applicazioni complete su. AWS In questo modello, si utilizza la CLI Amplify per distribuire le applicazioni micro-frontend Amplify.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire tramite comandi nella shell della riga di comando. Servizi AWS

Altri strumenti

- @angular -architects/module-federation è un plugin che integra Angular con Module Federation.
- Angular è un framework di applicazioni web open source per la creazione di applicazioni a pagina singola moderne, scalabili e testabili. Segue un'architettura modulare e basata su componenti che promuove il riutilizzo e la manutenzione del codice.
- Node.js è un ambiente di JavaScript runtime basato sugli eventi progettato per la creazione di applicazioni di rete scalabili.
- npm è un registro software che viene eseguito in un ambiente Node.js e viene utilizzato per condividere o prendere in prestito pacchetti e gestire la distribuzione di pacchetti privati.
- Webpack Module Federation ti aiuta a caricare codice compilato e distribuito in modo indipendente, come micro-frontend o plugin, in un'applicazione.

Archivio di codice

Il codice per questo pattern è disponibile nel portale Micro-frontend utilizzando il repository Angular e Module Federation. GitHub Questo repository contiene le due cartelle seguenti:

- `shell-app`contiene il codice per l'applicazione shell.
- `feature1-app`contiene un esempio di micro-frontend. L'applicazione shell recupera questo micro-frontend e lo visualizza come pagina all'interno dell'applicazione del portale.

Best practice

Le architetture Micro-frontend offrono numerosi vantaggi, ma introducono anche complessità. Di seguito sono riportate alcune best practice per uno sviluppo fluido, codice di alta qualità e un'ottima esperienza utente:

- Pianificazione e comunicazione: per semplificare la collaborazione, investi in pianificazione e progettazione iniziali e canali di comunicazione chiari.
- Coerenza del design: applica uno stile visivo coerente su tutti i microfrontend utilizzando sistemi di progettazione, guide di stile e librerie di componenti. Ciò fornisce un'esperienza utente coerente e accelera lo sviluppo.
- Gestione delle dipendenze: poiché i microfrontend si evolvono in modo indipendente, adottano contratti e strategie di controllo delle versioni standardizzati per gestire efficacemente le dipendenze e prevenire problemi di compatibilità.

- Architettura micro-frontend: per consentire lo sviluppo e l'implementazione indipendenti, ogni microfrontend dovrebbe avere una responsabilità chiara e ben definita per una funzionalità incapsulata.
- Integrazione e comunicazione: per facilitare l'integrazione e ridurre al minimo i conflitti, definisci contratti e protocolli di comunicazione chiari tra i microfrontend, inclusi eventi e modelli di dati condivisi. APIs
- Test e garanzia della qualità: implementa l'automazione dei test e le pipeline di integrazione continua per i microfrontend. Ciò migliora la qualità generale, riduce gli sforzi di test manuali e convalida la funzionalità tra le interazioni di microfrontend.
- Ottimizzazione delle prestazioni: monitora continuamente le metriche delle prestazioni e monitora le dipendenze tra i micro-frontend. Questo aiuta a identificare i punti deboli e a mantenere prestazioni ottimali delle applicazioni. A tale scopo, utilizzate strumenti di monitoraggio delle prestazioni e analisi delle dipendenze.
- Esperienza degli sviluppatori: concentrati sull'esperienza degli sviluppatori fornendo documentazione, strumenti ed esempi chiari. Questo ti aiuta a semplificare lo sviluppo e ad integrare nuovi membri del team.

Epiche

Crea l'applicazione shell

Operazione	Description	Competenze richieste

Crea l'applicazione shell.	1. Nella CLI di Angular, inserisci il seguente comando:
	`ng new shell --routing`
	2. Immettete il seguente comando per accedere alla cartella del progetto:
	`cd shell`
		Sviluppatore di app

Operazione	Description	Competenze richieste
	Note La struttura delle cartelle e dei progetti per le applicazioni shell e micro-frontend può essere totalment e indipendente. Possono essere gestite come applicazioni Angular indipendenti.	
Installa il plugin .	Nella CLI di Angular, inserisci il seguente comando per installare il plugin @angular - architects/module-federation: `ng add @angular-architects/module-federation --project shell --port 4200`	Sviluppatore di app

Operazione	Description	Competenze richieste
Aggiungi l'URL del micro-frontend come variabile di ambiente.	1. Apri il file environment.ts. 2. Aggiungi `mfe1URL: 'http://localhost:5000'` all'oggetto: `environment` `export const environment = { production: false, mfe1URL: 'http://localhost:5000', };` 3. Salva e chiudi il file environment.ts.	Sviluppatore di app

Operazione	Description	Competenze richieste

Definisci il routing.	1. Apri il file app-routing.module.ts.
	2. Nella CLI di Angular, inserisci il seguente comando per importare il `loadRemoteModule` modulo dal plugin @angular-architects/module-federation:
	`import { loadRemoteModule } from '@angular-architects/module-federation';`
	3. Imposta il percorso predefinito come segue:
	`{ path: '', pathMatch: 'full', redirectTo: 'mfe1' },`
	4. Imposta il percorso per il micro-frontend:
	`{ path: 'mfe1', loadChildren: () => loadRemoteModule({ type: 'module',`
		Sviluppatore di app

Operazione	Description	Competenze richieste
	<pre>remoteEntry: `${environment.mfe1URL}/remoteEntry.js`, exposedModule: './Module' }) .then(m => m.Mfe1Module) },</pre> 5. Salva e chiudi il file app-routing.module.ts.	
`mfe1`Dichiara il modulo.	1. Nella `src` cartella, crea un nuovo file denominato decl.d.ts. 2. Apri il file decl.d.ts. 3. Aggiungi quanto segue al file: <pre>declare module 'mfe1/Module';</pre> 4. Salvate e chiudete il file decl.d.ts.	Sviluppatore di app

Operazione	Description	Competenze richieste

Prepara il precaricamento per il micro-frontend.	Il precaricamento del microfrontend aiuta il webpack a negoziare correttamente le librerie e i pacchetti condivisi.
	1. Apri il file main.ts.
	2. Sostituisci il contenuto con quanto segue:
	`import { loadRemoteEntry } from '@angular-architects/module-federation';`
	`Promise.all([ loadRemoteEntry(` `${environment.mfe1URL}/remoteEntry.js` `, 'mfe1'), ]) .catch(err => console.error('Error loading remote entries', err)) .then(() => import('./bootstrap')) .catch(err => console.error(err) );`
	3. Salva e chiudi il file main.ts.
		Sviluppatore di app

Operazione	Description	Competenze richieste
Modifica il contenuto HTML.	1. Apri il file app.component.html. 2. Sostituisci il contenuto con quanto segue: `<h1>Shell application is running!</h1> <router-outlet></router-outlet>` 3. Salvate e chiudete il file app.component.html.	Sviluppatore di app

Crea l'applicazione micro-frontend

Operazione	Description	Competenze richieste

Crea il micro-frontend.	1. Nella CLI di Angular, inserisci il seguente comando:
	`ng new mfe1 --routing`
	2. Immettete il seguente comando per accedere alla cartella del progetto:
	`cd mfe1`
		Sviluppatore di app

Installa il plugin .	Immettete il seguente comando per installare il plugin @angular -architects/module-federation: `ng add @angular-architects/module-`	Sviluppatore di app

Operazione	Description	Competenze richieste
	`federation --project mfe1 --port 5000`	

Crea un modulo e un componente.	Immettete i seguenti comandi per creare un modulo e un componente ed esportarli come modulo di immissione remota:
	`ng g module mfe1 --routing`
	`ng g c mfe1`
		Sviluppatore di app

Imposta il percorso di routing predefinito.	1. Aprire il file mfe-routing.module.ts. 2. Imposta il percorso predefinito come segue: `{ path: '', component: Mfe1Component },` 3. Salvate e chiudete il file mfe-routing.module.ts.	Sviluppatore di app

Operazione	Description	Competenze richieste

`mfe1`Aggiungi il percorso.	1. Apri il file app-routing.module.ts.
	2. Imposta il percorso predefinito come segue:
	`{ path: '', pathMatch: 'full', redirectTo: 'mfe1' },`
	3. Aggiungi il seguente `mfe1` percorso:
	`{ path: 'mfe1', loadChildren: () => import('./mfe1/mfe1.module').then((m) => m.Mfe1Module), },`
	4. Salva e chiudi il file app-routing.module.ts.
		Sviluppatore di app

Operazione	Description	Competenze richieste

Modifica il file webpack.config.js.	1. Apri il file webpack.config.js.
	2. Modifica la `For remotes` sezione in modo che corrisponda a quanto segue:
	`// For remotes (please adjust)`
	`name: "mfe1",`
	`filename: "remoteEntry.js",`
	`exposes: {`
	`'./Module': './src/app/mfe1/mfe1.module.ts',`
	`},`
	3. Nella `shared` sezione, aggiungi tutte le dipendenze che l'`mfe1`applicazione condivide con l'applicazione shell:
	`shared: share({`
	`"@angular/core": { singleton: true, strictVersion: true, requiredVersion: 'auto' },`
	`"@angular/common": { singleton: true, strictVersion: true, requiredVersion: 'auto' },`
	`"@angular/common/http": { singleton: true, strictVersion: true,`
		Sviluppatore di app

Operazione	Description	Competenze richieste
	``` requiredVersion: 'auto' }, "@angular/ router": { singleton : true, strictVer sion: true, requiredV ersion: 'auto' }, ...shared Mappings.getDescri ptors() }) ``` 4. Salvate e chiudete il file webpack.config.js.	
Modifica il contenuto HTML.	1. Apri il file app.compo nent.html. 2. Sostituisci il contenuto con quanto segue: ``` <router-outlet></r outer-outlet> ``` 3. Salvate e chiudete il file app.component.html.	Sviluppatore di app

Esegui le applicazioni localmente

Operazione	Description	Competenze richieste
Esegui l'`mfe1`applicazione.	1. Inserisci il seguente comando per avviare l'`mfe1`applicazione: ``` npm start ```	Sviluppatore di app

Operazione	Description	Competenze richieste
	2. In un browser Web, accedihttp://localhost:5000 . 3. Verifica che il micro-frontend possa essere eseguito in modo autonomo. L'mfe1applicazione dovrebbe renderizzare correttamente senza errori.	
Esegui l'applicazione shell.	1. Immettete il seguente comando per avviare l'applicazione shell: `npm start` 2. In un browser Web, accedihttp://localhost:4200/mfe1 . 3. Verifica che il mfe1 micro-frontend sia incorporato nell'applicazione shell. L'applicazione del portale dovrebbe essere visualizzata correttamente senza errori e l'mfe1applicazione dovrebbe essere incorporata al suo interno.	Sviluppatore di app

Effettua il refactoring dell'applicazione shell per gestire un errore di caricamento del micro-frontend

Operazione	Description	Competenze richieste
Crea un modulo e un componente.	Nella cartella principal e dell'applicazione shell,	Sviluppatore di app

Operazione	Description	Competenze richieste

	immettete i seguenti comandi per creare un modulo e un componente per una pagina di errore:
	`ng g module error-page --routing`
	`ng g c error-page`	

Modifica il contenuto HTML.	1. Apri il file error-page.component.html. 2. Sostituisci il contenuto con quanto segue: `<p>Sorry, this page is not available.</p>` 3. Salva e chiudi il file error-page.component.html.	Sviluppatore di app
Imposta il percorso di routing predefinito.	1. Apri il file error-page-routing.module.ts. 2. Imposta il percorso predefinito come segue: `{ path: '', component: ErrorPageComponent },` 3. Salva e chiudi il error-page-routingfile .module.ts.	Sviluppatore di app

Operazione	Description	Competenze richieste

Crea una funzione per caricare i micro-frontend.	1. Apri il file app-routing.module.ts.
	2. Crea la seguente funzione:
	`function loadMFE(url: string) { return loadRemoteModule({ type: 'module', remoteEntry: `${url}/remoteEntry.js`, exposedModule: './Module' }) .then(m => m.Mfe1Module) .catch( () => import('./error-page/error-page.module').then(m => m.ErrorPageModule) ); }`
	3. Modifica il `mfe1` percorso in modo che sia il seguente:
	`{ path: 'mfe1', loadChildren: () => loadMFE(environment.mfe1URL) },`
	4. Salva e chiudi il file app-routing.module.ts.
		Sviluppatore di app

Operazione	Description	Competenze richieste

Verifica la gestione degli errori.	1. Se non è già in esecuzione, inserisci il seguente comando per avviare l'applicazione shell:
	`npm start`
	2. In un browser Web, accedi `http://localhost:4200/mfe1` .
	3. Verifica che la pagina di errore sia visualizzata. Dovresti vedere il testo seguente:
	`Sorry, this page is not available.`
		Sviluppatore di app

Distribuisci le applicazioni utilizzando AWS Amplify

Operazione	Description	Competenze richieste
Implementa il micro-frontend.	1. Nella CLI Amplify, accedi alla cartella principale dell'applicazione micro-frontend. 2. Inserisci il seguente comando per inizializzare Amplify: `amplify init` 3. Quando ti viene richiesto di inserire un nome per il	Sviluppatore di app, AWS DevOps

Operazione	Description	Competenze richieste
	tuo progetto Amplify, premi Invio. Questo riutilizza il nome dal file package.json. 4. Quando ti viene richiesto di inizializzare il progetto con la configurazione precedente, inserisci. `Yes` 5. Quando vi viene richiesto di selezionare un metodo di autenticazione, scegliete. `AWS Profile` 6. Seleziona il profilo che desideri utilizzare. 7. Attendi che Amplify inizializzi il progetto. Quando questo processo è completo, riceverai un messaggio di conferma nel terminale. 8. Inserisci il seguente comando per aggiungere una categoria di hosting Amplify al micro-frontend: `amplify add hosting` 9. Quando ti viene richiesto di selezionare il modulo del plugin, scegli. `Hosting with Amplify Console` 10. Quando ti viene richiesto di scegliere un tipo, scegli. `Manual deployment`	

Operazione	Description	Competenze richieste

	11. Installa le dipendenze npm del progetto inserendo il seguente comando:
	`npm install`
	12. Pubblica l'applicazione sulla console Amplify inserendo il seguente comando:
	`amplify publish -y`
	Una volta completata la pubblicazione, Amplify restituisce l'URL del micro-frontend.
	13. Copia l'URL. È necessario questo valore per aggiornar e l'applicazione shell.	

Operazione	Description	Competenze richieste

Distribuisci l'applicazione shell.	1. Nella src/app/environmentscartella, apri il file environments.prod.ts.
	2. Sostituisci il `mfe1URL` valore con l'URL del micro-frontend distribuito:
	`export const environment = { production: true, mfe1URL: 'https://<env>.<Amplify-app-ID>.amplifyapp.com' };`
	3. Salva e chiudi il file environments.prod.ts.
	4. Nella CLI Amplify, accedi alla cartella principale dell'applicazione shell.
	5. Inserisci il seguente comando per inizializzare Amplify:
	`amplify init`
	6. Quando ti viene richiesto di inserire un nome per il tuo progetto Amplify, premi Invio. Questo riutilizza il nome dal file package.json.
	7. Quando ti viene richiesto di inizializzare il progetto con la configurazione precedente, inserisci. `Yes`
		Sviluppatore dell'app, proprietario dell'app

Operazione	Description	Competenze richieste

	8. Quando vi viene richiesto di selezionare un metodo di autenticazione, scegliete. `AWS Profile`
	9. Seleziona il profilo che desideri utilizzare.
	10. Attendi che Amplify inizializzi il progetto. Quando questo processo è completo, riceverai un messaggio di conferma nel terminale.
	11. Aggiungi una categoria di hosting Amplify all'appli cazione shell:
	`amplify add hosting`
	12. Quando ti viene richiesto di selezionare il modulo del plugin, scegli. `Hosting with Amplify Console`
	13. Quando ti viene richiesto di scegliere un tipo, scegli. `Manual deployment`
	14. Installa le dipendenze npm del progetto inserendo il seguente comando:
	`npm install`
	15. Pubblica l'applicazione shell sulla console Amplify inserendo il seguente comando:	

Operazione	Description	Competenze richieste
	`amplify publish -y` Al termine della pubblicazione, Amplify restituisce l'URL dell'applicazione shell distribuita. 16. Prendi nota dell'URL dell'applicazione shell.	

Operazione	Description	Competenze richieste
Abilita CORS.	Poiché le applicazioni shell e microfrontend sono ospitate in modo indipendente su domini diversi, è necessari o abilitare la condivisione delle risorse tra le origini (CORS) sul microfrontend. Ciò consente all'applicazione shell di caricare il contenuto da un'origine diversa. Per abilitare CORS, aggiungete intestazioni personalizzate. 1. Nella CLI Amplify, accedi alla cartella principale del micro-frontend. 2. Immetti il comando seguente: `amplify configure hosting` 3. Quando ti viene richiesto di configurare le impostazioni personalizzate, inserisci. Y 4. Accedi a Console di gestione AWS, quindi apri la console Amplify. 5. Scegli il micro-frontend. 6. Nel riquadro di navigazione, scegli Hosting, quindi scegli Intestazioni personalizzate. 7. Scegli Modifica.	Sviluppatore di app, AWS DevOps

Operazione	Description	Competenze richieste

	8. Nella finestra Modifica intestazioni personalizzate, inserisci quanto segue:
	
	`customHeaders:`
	`  - pattern: '*.js'`
	`  headers:`
	`  - key: Access-Control-Allow-Origin`
	`  value: '*'`
	`  - key: Access-Control-Allow-Methods`
	`  value: 'GET, OPTIONS'`
	`  - key: Access-Control-Allow-Headers`
	`  value: '*'`
	
	9. Scegli Save (Salva).
	10. Ridistribuite il micro-frontend per applicare le nuove intestazioni personalizzate.	

Operazione	Description	Competenze richieste
Crea una regola di riscrittura sull'applicazione shell.	L'applicazione Angular shell è configurata per utilizzare il HTML5 routing. Se l'utente esegue un aggiornamento completo, Amplify tenta di caricare una pagina dall'URL corrente. Ciò genera un errore 403. Per evitare ciò, aggiungi una regola di riscrittura nella console Amplify. Per creare la regola di riscrittura, segui questi passaggi: 1. Nella CLI Amplify, accedi alla cartella principale dell'applicazione shell. 2. Immetti il comando seguente: `amplify configure hosting` 3. Quando ti viene richiesto di configurare le impostazioni personalizzate, inserisci. Y 4. Apri la console Amplify. 5. Scegli l'applicazione shell. 6. Nel riquadro di navigazione, scegli Hosting, quindi scegli Riscritture e reindirizzamenti. 7. Nella pagina Riscritture e reindirizzamenti, scegli Gestisci reindirizzamenti.	Sviluppatore di app, AWS DevOps

Operazione	Description	Competenze richieste
	8. Scegli Apri editor di testo. 9. Nell'editor JSON, inserisci il seguente reindirizzamento: `[ { "source": "/<*>", "target": "/index.html", "status": "404-200", "condition": null } ]` 10. Scegli Save (Salva).	
Prova il portale web.	1. In un browser Web, immettete l'URL dell'applicazione shell distribuita. 2. Verificate che l'applicazione shell e il micro-frontend vengano caricati correttamente.	Sviluppatore di app

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Eliminare le applicazioni.	Se non avete più bisogno delle applicazioni shell e micro-frontend, cancellatele. Questo aiuta a evitare addebiti per risorse che non utilizzate.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	1. Accedi a Console di gestione AWS, quindi apri la console Amplify. 2. Scegli il micro-frontend. 3. Nel riquadro di navigazione, scegli Impostazioni app, quindi scegli Impostazioni generali. 4. Scegli Elimina app. 5. Nella finestra di conferma, inserisci `delete`, quindi scegli Elimina app. 6. Ripeti questi passaggi per eliminare l'applicazione shell.	

risoluzione dei problemi

Problema	Soluzione
Nessun AWS profilo disponibile durante l'esecuzione del `amplify init` comando	Se non hai un AWS profilo configurato, puoi comunque procedere con il `amplify init` comando. Tuttavia, è necessario selezionare l'`AWS access key`opzione quando viene richiesto il metodo di autenticazione. Tieni a disposizione la chiave di AWS accesso e la chiave segreta. In alternativa, è possibile configurare un profilo denominato per AWS CLI. Per istruzioni, consulta Impostazioni dei file di configurazione e credenziali nella AWS CLI documentazione.

Problema	Soluzione
Errore durante il caricamento delle voci remote	Se riscontrate un errore durante il caricamento delle voci remote nel file main.ts dell'applicazione shell, assicuratevi che la `environment.mfe1URL` variabile sia impostata correttamente. Il valore di questa variabile deve essere l'URL del micro-frontend.
Errore 404 durante l'accesso al micro-frontend	Se ricevi un errore 404 quando provi ad accedere al micro-frontend locale, ad esempio all'indirizzo`http://localhost:4200/mfe1`, controlla quanto segue: • Per l'applicazione shell, assicurati che la configurazione di routing nel file app-routing.module.ts sia impostata correttamente e assicurati che la funzione stia chiamando correttamente il micro-frontend. `loadRemoteModule` • Per il micro-frontend, verificate che il file webpack.config.js abbia la configurazione corretta e assicuratevi che il file remoteEntry.js sia generato correttamente. **`exposes`**

Informazioni aggiuntive

AWS documentazione

- Comprensione e implementazione dei microfrontend su AWS(Prescriptive Guidance)AWS
- CLI Amplify (documentazione Amplify)
- Amplify Hosting (documentazione Amplify)

Altri riferimenti

- Federazione dei moduli

- Node.js
- Angolare
- @angular -architects/module-federation

Implementa un'applicazione a pagina singola basata su React su Amazon S3 e CloudFront

Jean-Baptiste Guillois, Amazon Web Services

Riepilogo

Un'applicazione a pagina singola (SPA) è un sito Web o un'applicazione Web che aggiorna dinamicamente il contenuto di una pagina Web visualizzata utilizzando. JavaScript APIs Questo approccio migliora l'esperienza utente e le prestazioni di un sito Web poiché aggiorna solo i nuovi dati anziché ricaricare l'intera pagina Web dal server.

Questo modello fornisce un step-by-step approccio alla codifica e all'hosting di una SPA scritto in React su Amazon Simple Storage Service (Amazon S3) e Amazon. CloudFront La SPA in questo modello utilizza un'API REST configurata in Amazon API Gateway ed esposta tramite una CloudFront distribuzione Amazon per semplificare la gestione della condivisione delle risorse tra le origini (CORS).

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Node.js enpm, installato e configurato. Per ulteriori informazioni, consulta la sezione Download della documentazione di Node.js.
- Yarn, installato e configurato. Per ulteriori informazioni, consulta la documentazione di Yarn.
- Git, installato e configurato. Per ulteriori informazioni, consulta la documentazione di Git.

Architecture

Questa architettura viene implementata automaticamente utilizzando AWS CloudFormation (infrastruttura come codice). Utilizza servizi regionali come Amazon S3 per archiviare gli asset statici e Amazon CloudFront con Amazon API Gateway per esporre gli endpoint API regionali (REST). I log delle applicazioni vengono raccolti utilizzando Amazon CloudWatch. Tutte le chiamate AWS API vengono verificate. AWS CloudTrail Tutte le configurazioni di sicurezza (ad esempio, identità e autorizzazioni) sono gestite in AWS Identity and Access Management (IAM). I contenuti statici

vengono distribuiti tramite Amazon CloudFront Content Delivery Network (CDN) e le query DNS vengono gestite da Amazon Route 53.

Tools (Strumenti)

Servizi AWS

- Amazon API Gateway ti aiuta a creare, pubblicare, gestire, monitorare e proteggere REST, HTTP e WebSocket APIs su qualsiasi scala.
- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita in tutte Account AWS le regioni.
- Amazon CloudFront accelera la distribuzione dei tuoi contenuti web distribuendoli attraverso una rete mondiale di data center, che riduce la latenza e migliora le prestazioni.
- AWS CloudTrailti aiuta a controllare la governance, la conformità e il rischio operativo del tuo. Account AWS
- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue AWS risorse e delle applicazioni su cui esegui AWS in tempo reale.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon Route 53 è un servizio Web DNS altamente scalabile e disponibile.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Codice

Il codice applicativo di esempio di questo pattern è disponibile nell'archivio di applicazioni CORS a pagina singola GitHub basato su React.

Best practice

Utilizzando lo storage di oggetti Amazon S3, puoi archiviare gli asset statici dell'applicazione in modo sicuro, altamente resiliente, performante ed economico. Non è necessario utilizzare un contenitore dedicato o un'istanza Amazon Elastic Compute Cloud (Amazon EC2) per questa attività.

Utilizzando la rete di distribuzione di CloudFront contenuti Amazon, puoi ridurre la latenza che i tuoi utenti potrebbero riscontrare quando accedono alla tua applicazione. Puoi anche collegare un firewall per applicazioni Web (AWS WAF) per proteggere le tue risorse da attacchi dannosi.

Epiche

Crea e distribuisci localmente la tua applicazione

Operazione	Description	Competenze richieste
Clonare il repository.	Eseguite il comando seguente per clonare il repository dell'applicazione di esempio: `git clone https://github.com/aws-samples/react-cors-spa react-cors-spa && cd react-cors-spa`	Sviluppatore di app, AWS DevOps
Distribuisci l'applicazione localmente.	1. Nella directory del progetto, esegui il `npm install` comando per avviare le dipendenze dell'applicazione. 2. Esegui il `yarn dev` comando per avviare l'applicazione localmente.	Sviluppatore di app, AWS DevOps
Accedi localmente all'applicazione.	Apri una finestra del browser e inserisci l'`http://localhost:3000` URL per accedere all'applicazione.	Sviluppatore di app, AWS DevOps

Distribuzione dell’applicazione

Operazione	Description	Competenze richieste
Implementa il AWS CloudFormation modello.	1. Accedi a Console di gestione AWS, quindi apri	Sviluppatore di app, AWS DevOps

Operazione	Description	Competenze richieste
	la AWS CloudFormation console. 2. Scegli Crea stack, quindi scegli Con nuove risorse (standard). 3. Scegliere Upload a template file (Carica un file di modello). 4. Scegli il file, scegli il **`react-cors-spa-stack.yaml`** file dal repository clonato, quindi scegli Avanti. 5. Inserisci un nome per lo stack, quindi scegli Avanti. 6. Mantieni tutte le opzioni predefinite, quindi scegli Avanti. 7. Controlla le impostazioni finali dello stack, quindi scegli Crea pila.	

Operazione	Description	Competenze richieste
Personalizza i file sorgente dell'applicazione.	1. Dopo aver distribuito lo stack, apri la scheda Output e identifica il `Bucket` nome e `APIDomain` il valore. 2. Copia il dominio CloudFront di distribuzione per l'API REST. 3. Vai a `<project_root>/src/pages/index.tsx`, quindi inserisci o incolla questo dominio nel valore della `APIEndPoint` variabile alla riga 13 del `index.tsx` file.	Sviluppatore di app
Compila il pacchetto applicativo.	Nella directory del progetto, esegui il `yarn build` comando per creare il pacchetto dell'applicazione.	Sviluppatore di app

Operazione	Description	Competenze richieste
Distribuisci il pacchetto dell'applicazione.	1. Apri la console Amazon S3. 2. Identifica e scegli il bucket S3 creato in precedenza dallo stack. CloudFormation 3. Scegli Carica, quindi scegli Aggiungi file. 4. Scegli il contenuto della tua `out` cartella. 5. Scegli Aggiungi cartella, quindi scegli la `_next` directory. **Important** Scegli la `_next` cartella, non il contenuto. 6. Scegli Carica per caricare i file e la directory nel tuo bucket S3.	Sviluppatore di app, AWS DevOps

Eseguire il test dell'applicazione

Operazione	Description	Competenze richieste
Accedere e testare l'applica zione.	Apri una finestra del browser, quindi incolla il dominio di CloudFront distribuz ione (l'`SPADomain` output CloudFormation dello stack distribuito in precedenza) per accedere all'applicazione.	Sviluppatore di app, AWS DevOps

Pulisci le risorse

Operazione	Description	Competenze richieste
Elimina il contenuto del bucket S3.	1. Apri la console Amazon S3 e scegli il bucket creato in precedenza dallo stack (il primo bucket il cui nome inizia con). `react-cors-spa-` 2. Scegli Empty per eliminare il contenuto del bucket. 3. Scegli il secondo bucket creato in precedenza dallo stack (il secondo bucket il cui nome inizia con `react-cors-spa-` e finisce con). `-logs` 4. Scegli Vuoto per eliminare il contenuto del bucket.	AWS DevOps, sviluppatore di app
Elimina lo CloudFormation stack.	1. Apri la CloudFormation console e scegli lo stack che hai creato in precedenza. 2. Scegli Elimina per eliminare lo stack e tutte le risorse correlate.	AWS DevOps, sviluppatore di app

Risorse correlate

Per distribuire e ospitare la tua applicazione web, puoi anche utilizzare AWS Amplify Hosting, che fornisce un flusso di lavoro basato su Git per ospitare app web serverless complete con distribuzione continua. Amplify Hosting fa parte AWS Amplifydi, che fornisce una serie di strumenti e funzionalità appositamente progettati che consentono agli sviluppatori web e mobili frontend di creare rapidamente e facilmente applicazioni complete su. AWS

Informazioni aggiuntive

Per gestire le URLs richieste non valide dell'utente che potrebbero generare errori 403, una pagina di errore personalizzata configurata nella CloudFront distribuzione rileva gli errori 403 e li reindirizza al punto di ingresso dell'applicazione (). `index.html`

Per semplificare la gestione di CORS, l'API REST viene esposta tramite una distribuzione. CloudFront

Implementa un'API Amazon API Gateway su un sito Web interno utilizzando endpoint privati e un Application Load Balancer

Saurabh Kothari, Amazon Web Services

Riepilogo

Questo modello mostra come implementare un'API Amazon API Gateway su un sito Web interno accessibile da una rete locale. Imparerai a creare un nome di dominio personalizzato per un'API privata utilizzando un'architettura progettata con endpoint privati, Application Load Balancer, PrivateLink AWS e Amazon Route 53. Questa architettura previene le conseguenze indesiderate dell'utilizzo di un nome di dominio e di un server proxy personalizzati per facilitare il routing basato sul dominio su un'API. Ad esempio, se si implementa un endpoint di cloud privato virtuale (VPC) in una sottorete non instradabile, la rete non può raggiungere API Gateway. Una soluzione comune consiste nell'utilizzare un nome di dominio personalizzato e quindi distribuire l'API in una sottorete instradabile, ma ciò può interrompere altri siti interni quando la configurazione del proxy trasferisce il traffico (`execute-api.{region}.vpce.amazonaws.com`) ad AWS Direct Connect. Infine, questo modello può aiutarti a soddisfare i requisiti organizzativi per l'utilizzo di un'API privata non raggiungibile da Internet e di un nome di dominio personalizzato.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un certificato SNI (Server Name Indication) per il tuo sito Web e la tua API
- Una connessione da un ambiente locale a un account AWS configurato utilizzando AWS Direct Connect o AWS VPN Site-to-Site
- Una zona ospitata privata con un dominio corrispondente (ad esempio domain.com) che viene risolta da una rete locale e inoltra le query DNS a Route 53
- Una sottorete privata instradabile raggiungibile da una rete locale

Limitazioni

Per ulteriori informazioni sulle quote (precedentemente denominate limiti) per i bilanciamenti del carico, le regole e altre risorse, consulta Quotas for your Application Load Balancers nella documentazione di Elastic Load Balancing.

Architecture

Stack tecnologico

- Gateway Amazon API
- Amazon Route 53
- Application Load Balancer
- AWS Certificate Manager
- AWS PrivateLink

Architettura Target

Il diagramma seguente mostra come viene distribuito un Application Load Balancer in un VPC che indirizza il traffico Web verso un gruppo target di siti Web o un gruppo target API Gateway in base alle regole del listener Application Load Balancer. Il gruppo target API Gateway è un elenco di indirizzi IP per l'endpoint VPC in API Gateway. API Gateway è configurato per rendere l'API privata con la relativa politica delle risorse. La policy rifiuta tutte le chiamate che non provengono da un endpoint VPC specifico. I nomi di dominio personalizzati in API gateway vengono aggiornati per utilizzare api.domain.com per l'API e la relativa fase. Le regole Application Load Balancer vengono aggiunte al traffico di routing in base al nome host.

Il diagramma mostra il flusso di lavoro seguente:

1. Un utente di una rete locale tenta di accedere a un sito Web interno. La richiesta viene inviata a ui.domain.com e api.domain.com. La richiesta viene quindi risolta nell'Application Load Balancer interno della sottorete privata instradabile. L'SSL viene terminato all'Application Load Balancer per ui.domain.com e api.domain.com.
2. Le regole del listener, configurate sull'Application Load Balancer, controllano l'intestazione dell'host.

   a. Se l'intestazione host è api.domain.com, la richiesta viene inoltrata al gruppo target API Gateway. L'Application Load Balancer avvia una nuova connessione all'API Gateway tramite la porta 443.

   b. Se l'intestazione dell'host è ui.domain.com, la richiesta viene inoltrata al gruppo di destinazione del sito Web.

3. Quando la richiesta raggiunge API Gateway, la mappatura personalizzata del dominio configurata in API Gateway determina il nome host e l'API da eseguire.

Automazione e scalabilità

I passaggi di questo modello possono essere automatizzati utilizzando AWS CloudFormation o AWS Cloud Development Kit (AWS CDK). Per configurare il gruppo di destinazione delle chiamate API Gateway, è necessario utilizzare una risorsa personalizzata per recuperare l'indirizzo IP dell'endpoint VPC. Chiama describe-vpc-endpointse describe-network-interfacesrestituisce gli indirizzi IP e il gruppo di sicurezza, che possono essere utilizzati per creare il gruppo target di indirizzi IP dell'API.

Tools (Strumenti)

- Amazon API Gateway ti aiuta a creare, pubblicare, gestire, monitorare e proteggere REST, HTTP e WebSocket APIs su qualsiasi scala.
- Amazon Route 53 è un servizio Web DNS altamente scalabile e disponibile.
- AWS Certificate Manager (ACM) ti aiuta a creare, archiviare e rinnovare certificati e chiavi SSL/TLS X.509 pubblici e privati che proteggono i tuoi siti Web e le tue applicazioni AWS.
- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software che aiuta a definire e fornire l'infrastruttura cloud AWS in codice.
- AWS ti PrivateLink aiuta a creare connessioni private unidirezionali dai tuoi VPCs servizi esterni al VPC.

Epiche

Crea un certificato SNI

Operazione	Description	Competenze richieste
Crea un certificato SNI e importalo in ACM.	1. Crea un certificato SNI per ui.domain.com e api.domain.com. Per ulteriori informazioni, consulta Scelta del modo in cui CloudFront vengono servite le richieste HTTPS nella	Amministratore di rete

Operazione	Description	Competenze richieste
	CloudFront documenta zione di Amazon. 2. Importa i certificati SNI in AWS Certificate Manager (ACM). Per ulteriori informazioni, consulta Importazione di certificati in AWS Certificate Manager nella documentazione ACM.	

Implementa un endpoint VPC in una sottorete privata non instradabile

Operazione	Description	Competenze richieste
Crea un endpoint VPC di interfaccia in API Gateway.	Per creare un endpoint VPC di interfaccia, segui le istruzion i di Accedere a un servizio AWS utilizzando un endpoint VPC di interfaccia nella documentazione di Amazon Virtual Private Cloud (Amazon VPC).	Amministratore del cloud

Configurazione dell'Application Load Balancer

Operazione	Description	Competenze richieste
Crea un gruppo target per la tua candidatura.	Crea un gruppo target per le risorse dell'interfaccia utente della tua applicazione.	Amministratore cloud
Crea un gruppo target per l'endpoint API Gateway.	1. Crea un gruppo target con un tipo di indirizzo IP, quindi aggiungi l'indirizzo	Amministratore cloud

Operazione	Description	Competenze richieste
	IP dell'endpoint VPC per l'endpoint API Gateway al gruppo di destinazione. 2. Configura i controlli di integrità per i tuoi gruppi target con il codice di successo 403. 403 è necessario perché l'endpoint VPC per l'API Gateway restituisce un codice 403 quando viene richiamato senza intestazioni dal controllo dello stato del gruppo target.	
Crea un Application Load Balancer.	1. Crea un Application Load Balancer (interno) in una sottorete privata instradabile. 2. Aggiungi il listener 443 all'Application Load Balancer e scegli il certificato da ACM.	Amministratore cloud
Crea regole per gli ascoltatori.	Create regole per gli ascoltatori per effettuare le seguenti operazioni: 1. Inoltra l'host api.domain.com al gruppo target API Gateway 2. Inoltra l'host ui.domain.com al gruppo di destinazione per le risorse dell'interfaccia utente	Amministratore cloud

Configura Route 53

Operazione	Description	Competenze richieste
Crea una zona ospitata privata.	Crea una zona ospitata privata per domain.com.	Amministratore cloud
Crea record di dominio.	Crea record CNAME per quanto segue: • Un'API con il valore impostato sul nome DNS dell'Application Load Balancer • Un'interfaccia utente con il valore impostato sul nome DNS dell'Application Load Balancer	Amministratore cloud

Crea un endpoint API privato in API Gateway

Operazione	Description	Competenze richieste
Crea e configura un endpoint API privato.	1. Per creare un endpoint API privato, segui le istruzioni riportate in Creazione di un'API privata in Amazon API Gateway nella documentazione di API Gateway. 2. Configura la politica delle risorse per consentire le chiamate solo all'API dall'endpoint VPC. Per ulteriori informazioni, consulta Controllare	Sviluppatore di app, amministratore cloud

Operazione	Description	Competenze richieste
	l'accesso a un'API con le policy delle risorse di API Gateway nella documenta zione di API Gateway.	
Crea un nome di dominio personalizzato.	1. Crea un nome di dominio personalizzato per api.domain.com. Per ulteriori informazioni, consulta Configurazione di nomi di dominio personali zzati per REST APIs nella documentazione di API Gateway. 2. Seleziona l'API e lo stage creati. Per ulteriori informazi oni, consulta Lavorare con le mappature delle API per REST APIs nella documentazione di API Gateway.	Amministratore cloud

Risorse correlate

- Gateway Amazon API
- Amazon Route 53
- Application Load Balancer
- AWS PrivateLink
- AWS Certificate Manager

Incorpora i componenti visivi di Amazon Quick Sight nelle applicazioni Web utilizzando Amazon Cognito e l'automazione IAc

Ishita Gupta, Saurabh Singh e Srishti Wadhwa, Amazon Web Services

Riepilogo

Questo modello offre un approccio specializzato per incorporare i componenti visivi di Amazon Quick Sight nelle applicazioni React utilizzando l'incorporamento di utenti registrati con l'autenticazione semplificata di Amazon Cognito. Queste risorse vengono quindi distribuite tramite un modello di infrastruttura come codice (IaC). A differenza del tradizionale incorporamento di dashboard, questa soluzione isola tabelle e grafici specifici per l'integrazione diretta nelle applicazioni React, il che migliora notevolmente sia le prestazioni che l'esperienza utente.

L'architettura stabilisce un flusso di autenticazione efficiente tra la gestione degli utenti di Amazon Cognito e le autorizzazioni Quick Sight: gli utenti si autenticano tramite Amazon Cognito e accedono alle visualizzazioni autorizzate in base alle regole di condivisione del dashboard in Quick Sight. Questo approccio semplificato elimina la necessità di un accesso diretto alla console Quick Sight, pur mantenendo solidi controlli di sicurezza.

L'ambiente completo viene distribuito tramite un unico AWS CloudFormation modello che fornisce tutti i componenti dell'infrastruttura necessari, tra cui:

- Un backend serverless che utilizza Amazon API AWS Lambda Gateway
- Hosting frontend sicuro tramite Amazon CloudFront, Amazon Simple Storage Service (Amazon S3) e AWS WAF
- Gestione delle identità tramite Amazon Cognito

Tutti i componenti sono configurati seguendo le migliori pratiche di sicurezza con politiche, protezione e crittografia dei privilegi minimi AWS Identity and Access Management (IAM). AWS WAF end-to-end

Questa soluzione è ideale per i team di sviluppo e le organizzazioni che desiderano integrare analisi sicure e interattive nelle proprie applicazioni, mantenendo al contempo un controllo granulare sull'accesso degli utenti. La soluzione utilizza servizi AWS gestiti e automazione per semplificare il processo di integrazione, migliorare la sicurezza e garantire la scalabilità per soddisfare le esigenze aziendali in evoluzione.

Destinatari e casi d'uso:

- Sviluppatori di frontend che desiderano incorporare l'analisi nelle app React
- Team di prodotto Software as a Service (SaaS) che desiderano offrire visualizzazioni dei dati per utente o basate sui ruoli
- Architetti di soluzioni interessati a integrare l'analisi in portali personalizzati AWS
- Sviluppatori di business intelligence (BI) che desiderano esporre le immagini agli utenti autenticati senza richiedere l'accesso completo alla dashboard
- Team aziendali che desiderano incorporare grafici Quick Sight interattivi negli strumenti interni

Prerequisiti e limitazioni

Prerequisiti

Per implementare con successo questo modello, assicurati che siano presenti le seguenti condizioni:

- Attivo Account AWS: Account AWS con autorizzazioni per distribuire CloudFormation stack e creare risorse Lambda, API Gateway, Amazon Cognito e Amazon S3 CloudFront.
- Account Amazon Quick Sight: un account Quick Sight attivo con almeno una dashboard contenente immagini. Per istruzioni di configurazione, consulta Tutorial: Creazione di un pannello di controllo Amazon Quick Sight utilizzando dati di esempio nella documentazione di Amazon Quick.
- Un ambiente di sviluppo composto da:
  - Node.js (versione 16 o successiva)
  - npm o yarn installato
  - Vite come strumento di compilazione di React
  - React (versione 19.1.1)
- Condivisione della dashboard: le dashboard devono essere condivise in Quick Sight e l'implementatore deve effettuare il login per accedere alle immagini o ai dashboard incorporati.

Limitazioni

- Questo modello utilizza il metodo di incorporamento degli utenti registrati, che richiede agli implementatori di disporre di un account Quick Sight attivo.
- L'accesso è limitato ai dashboard e agli elementi visivi che vengono condivisi esplicitamente con l'utente Quick Sight autenticato che implementa questo modello. Se l'implementatore non dispone dei diritti di accesso corretti, la generazione dell'URL di incorporamento avrà esito negativo e gli elementi visivi non verranno caricati.

- Lo CloudFormation stack deve essere distribuito in un ambiente in Regione AWS cui siano supportati Quick Sight, API Gateway e Amazon Cognito. Per la disponibilità per regione, consulta Servizi AWS per regione.

Versioni del prodotto

- Quick Sight Embedding SDK versione 2.10.1
- React versione 19.1.1
- Node.js versione 16 o successiva per garantire la compatibilità con le ultime versioni di React e Vite utilizzate in questa soluzione

Architecture

Architettura Target

Il diagramma seguente mostra l'architettura e il flusso di lavoro per questo modello.

In questo flusso di lavoro:

1. L'utente accede all'applicazione. L'utente apre l'applicazione web React utilizzando un browser. La richiesta viene indirizzata a una CloudFront distribuzione, che funge da rete di distribuzione dei contenuti per l'applicazione.
2. AWS WAF filtra le richieste dannose. Prima che la richiesta arrivi CloudFront, passa AWS WAF. AWS WAF ispeziona il traffico e blocca eventuali richieste dannose o sospette in base alle regole di sicurezza.
3. Amazon S3 serve file statici. Se la richiesta è pulita, CloudFront recupera le risorse statiche del frontend (HTML, JS, CSS) da un bucket S3 privato utilizzando Origin Access Control (OAC) e le invia al browser.
4. L'utente accede. Dopo il caricamento dell'applicazione, l'utente accede tramite Amazon Cognito, che autentica l'utente e restituisce un token web JSON (JWT) sicuro per l'accesso all'API autorizzato.
5. L'applicazione effettua una richiesta API. Dopo il login, l'applicazione React effettua una chiamata sicura all'`/get-embed-url`endpoint su API Gateway e passa il token JWT nell'intestazione della richiesta per l'autenticazione.

6. Il token è convalidato. API Gateway convalida il token utilizzando un autorizzatore Amazon Cognito. Se il token è valido, la richiesta procede; in caso contrario, viene rifiutata con una risposta 401 (non autorizzata).
7. La richiesta viene indirizzata a Lambda per l'elaborazione. La richiesta convalidata viene quindi inoltrata a una funzione Lambda di backend. Questa funzione è responsabile della generazione dell'URL di incorporamento per l'immagine Quick Sight richiesta.
8. Lambda genera l'URL di incorporamento da Quick Sight. IAM utilizza un ruolo IAM con le autorizzazioni appropriate per chiamare l'`GenerateEmbedUrlForRegisteredUser`API Quick Sight e generare un URL visivo sicuro e con ambito utente.
9. Lambda restituisce l'URL di incorporamento ad API Gateway. Lambda invia l'URL di incorporamento generato ad API Gateway come parte di una risposta JSON. Questa risposta viene quindi preparata per essere inviata al frontend.
10. L'URL di incorporamento viene inviato al browser. L'URL di incorporamento viene restituito al browser come risposta API.
11. L'immagine viene mostrata all'utente. L'applicazione React riceve la risposta e utilizza l'SDK Quick Sight Embedding per rendere l'immagine specifica all'utente.

Automazione e scalabilità

Le implementazioni di backend e frontend sono completamente automatizzate utilizzando CloudFormation, che fornisce tutte le AWS risorse richieste, tra cui Amazon Cognito, Lambda, API Gateway, Amazon S3,, ruoli IAM e Amazon in un' AWS WAF unica implementazione. CloudFront CloudWatch

Questa automazione garantisce un'infrastruttura coerente e ripetibile in tutti gli ambienti. Tutti i componenti si scalano automaticamente: Lambda si adatta alle chiamate di funzione, CloudFront fornisce i contenuti memorizzati nella cache a livello globale e API Gateway si adatta alle richieste in arrivo.

Tools (Strumenti)

Servizi AWS

- Amazon Quick Sight è un servizio di business intelligence nativo per il cloud che ti aiuta a creare, gestire e incorporare dashboard e immagini interattive.
- I gestori di Amazon API Gateway APIs fungono da ponte tra l'applicazione React e i servizi di backend.

- AWS Lambdaè un servizio di elaborazione serverless che questo modello utilizza per generare un incorporamento sicuro di Quick Sight in URLs modo dinamico e scalabile automaticamente in base alle richieste.
- Amazon Cognito fornisce l'autenticazione e l'autorizzazione per gli utenti ed emette token sicuri per l'accesso alle API.
- Amazon S3 ospita risorse frontend statiche per questo modello e le fornisce in modo sicuro. CloudFront
- Amazon CloudFront distribuisce contenuti frontend a livello globale con bassa latenza e si integra con AWS WAF il filtraggio del traffico.
- AWS WAFprotegge l'applicazione Web dal traffico dannoso applicando regole di sicurezza e limitando la velocità.
- AWS CloudFormationautomatizza il provisioning e la configurazione di tutte le risorse applicative in un'unica implementazione.
- Amazon CloudWatch raccoglie log e parametri da Lambda, API Gateway e per il monitoraggio AWS WAF e la risoluzione dei problemi.

Strumenti di sviluppo

- React JS è un framework di frontend che questo modello utilizza per creare l'applicazione web e integrare gli elementi visivi Quick Sight incorporati.
- Vite è uno strumento di compilazione utilizzato per lo sviluppo rapido e le build di produzione ottimizzate dell'applicazione React.
- Quick Sight Embedding SDK facilita l'incorporamento di elementi visivi Quick Sight nell'applicazione React e consente un'interazione perfetta tra l'applicazione e le immagini.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub Amazon Quick Sight Visual Embedding in React.

Best practice

Questo modello implementa automaticamente le seguenti best practice di sicurezza:

- Utilizza i pool di utenti di Amazon Cognito per l'autenticazione basata su JWT, con autenticazione a più fattori (MFA) opzionale.

- Protegge APIs con gli autorizzatori di Amazon Cognito e applica le politiche IAM con privilegi minimi su tutti i servizi.
- Implementa l'incorporamento degli utenti registrati Quick Sight e assegna automaticamente agli utenti il ruolo di lettore.
- Applica la crittografia in transito che supporta TLS 1.2 e versioni successive tramite HTTPS. CloudFront
- Crittografa i dati inattivi utilizzando AES-256 per Amazon S3 con controllo delle versioni e OAC.
- Configura i piani di utilizzo di API Gateway con throttling e quote.
- Protegge Lambda con concorrenza riservata e protezione delle variabili d'ambiente.
- Abilita la registrazione per Amazon S3 CloudFront, Lambda e API Gateway; monitora i servizi utilizzando. CloudWatch
- Crittografa i log, applica i controlli di accesso e applica politiche di negazione per i caricamenti non HTTPS o non crittografati.

Inoltre, consigliamo quanto segue:

- Utilizzalo CloudFormation per automatizzare le implementazioni e mantenere configurazioni coerenti tra gli ambienti.
- Assicurati che ogni utente disponga delle autorizzazioni Quick Sight corrette per accedere agli elementi visivi incorporati.
- Proteggi gli endpoint API Gateway con gli autorizzatori di Amazon Cognito e applica il principio del privilegio minimo per tutti i ruoli IAM.
- Archivia informazioni sensibili come Amazon Resource Names (ARNs) e IDs in variabili di ambiente anziché codificarle.
- Ottimizza le funzioni Lambda riducendo le dipendenze e migliorando le prestazioni di avviamento a freddo. Per ulteriori informazioni, consulta il post AWS sul blog Ottimizzazione delle prestazioni di avviamento a freddo dell' AWS Lambda utilizzo di strategie di innesco avanzate con. SnapStart
- Aggiungi il tuo CloudFront dominio all'elenco delle autorizzazioni di Quick Sight per abilitare l'incorporamento visivo sicuro.
- Monitora le prestazioni e la sicurezza utilizzando CloudWatch e AWS WAF per la registrazione, gli avvisi e la protezione del traffico.

Altre best practice consigliate

- Utilizza domini personalizzati con certificati SSL AWS Certificate Manager per fornire un'esperienza utente sicura e personalizzata.
- Crittografa i dati CloudWatch e i log di Amazon S3 utilizzando chiavi AWS Key Management Service gestite dal cliente AWS KMS() per un maggiore controllo sulla crittografia.
- Estendi AWS WAF le regole con geo-blocking, SQL injection (SQLi), protezione Cross-Site Scripting (XSS) e filtri personalizzati per una maggiore prevenzione delle minacce.
- Attivate gli CloudWatch allarmi e garantite il monitoraggio AWS Config, il controllo e AWS CloudTrail la conformità della configurazione in tempo reale.
- Applica politiche IAM granulari, applica la rotazione delle chiavi API e consenti l'accesso tra account solo quando assolutamente necessario.
- Esegui valutazioni di sicurezza regolari per garantire l'allineamento con i framework di conformità come System and Organization Controls 2 (SOC 2), il Regolamento generale sulla protezione dei dati (GDPR) e l'Health Insurance Portability and Accountability Act (HIPAA).

Epiche

Prepara l'ambiente

Operazione	Description	Competenze richieste

Clonare il repository.	Clona il GitHub repository di questa soluzione sul tuo sistema locale e vai alla directory del progetto:
	`git clone https://github.com/aws-samples/sample-quicksight-visual-embedding.git`
	`cd sample-quicksight-visual-embedding`
	Questo repository contiene il CloudFormation modello
		Sviluppatore di app

Operazione	Description	Competenze richieste
	e il codice sorgente React necessari per distribuire la soluzione.	

Implementa lo stack CloudFormation

Operazione	Description	Competenze richieste
Implementa il modello.	1. Accedi a Console di gestione AWS e apri la AWS CloudFormation console. 2. Nella pagina Pile, scegli Crea pila in alto a destra, quindi scegli Con nuove risorse (standard). 3. Nella pagina Crea stack, per Prerequisito — Prepara modello, scegli Scegli un modello esistente. 4. In Specify template (Specifica il modello), scegliere Upload a template file (Carica un file modello). 5. Carica il **`template.yaml`** file dal tuo GitHub repository locale, quindi scegli Avanti. 6. Nella pagina Specificare i dettagli dello stack, inserisci il nome dello stack (ad esempio,). `quicksight-embedding-stack`	Amministratore AWS

Operazione	Description	Competenze richieste
	7. Nella pagina Configura le opzioni dello stack, mantieni le impostazioni predefinite e seleziona la casella per confermare la creazione di risorse IAM. Scegli Next (Successivo). 8. Nella pagina Rivedi e crea, scegli Invia. Per ulteriori informazioni, consulta Creazione e gestione degli stack nella CloudFormation documentazione.	
Monitora la creazione di stack.	Monitora lo stack nella scheda Eventi fino a quando il suo stato non è CREATE_COMPLETE.	Amministratore AWS
Recupera gli output dello stack.	1. Quando la distribuzione è completa, scegli la scheda Output. 2. Nota gli output principali:`ApiGatewayUrl`, `UserPoolId` `UserPoolClientId` `CloudFrontDomainName`, e `S3BucketName`. Utilizzerai queste informazioni nei passaggi successivi.	Amministratore AWS

Configura l'ambiente frontend

Operazione	Description	Competenze richieste
Recupera gli identificatori visivi Quick Sight.	1. Apri la dashboard di Quick Sight. 2. Annota l'ID della dashboard, l'ID del foglio e l'ID visivo di ogni elemento visivo che intendi incorporare nella tua applicazione web.	Amministratore Quick Sight
Configura il tuo ambiente React locale.	Per configurare il tuo ambiente React locale e collegarlo alle AWS risorse, crea un `.env` file nella `my-app/` cartella del tuo GitHub repository locale. Compila il file con: • Il tuo Regione AWS • Informazioni sul pool di Amazon Cognito (dagli CloudFormation stack output) • Endpoint API Gateway (dagli output CloudForm ation dello stack) • Visualizzazione rapida (, , IDs) `DashboardId` `SheetId` `VisualId` Ecco un `.env` file di esempio: `VITE_AWS_REGION=us-east-1`	Sviluppatore di app

Operazione	Description	Competenze richieste

	`# Cognito Configura tion (from CloudForm ation outputs) VITE_USER_POOL_ID= us-east-1_xxxxxxxx x VITE_USER_POOL_WEB _CLIENT_ID=xxxxxxx xxxxxxxxxxxxxxxxxx`
	
	`# API Configuration (from CloudFormation outputs) VITE_API_URL=https :/your-api-id.exec ute-api.us-east-1. amazonaws.com/prod`
	
	`# QuickSight Visual Configuration VITE_DASHBOARD_ID=your -dashboard-id VITE_SHEET_ID=your- sheet-id VITE_VISUAL_ID=your- visual-id`	

Configura l'autenticazione degli utenti

Operazione	Description	Competenze richieste
Creare o gestire utenti in Cognito	Per consentire l'accesso degli utenti autenticati agli elementi visivi Quick Sight incorpora ti, devi prima creare utenti in Amazon Cognito: 1. Sulla console Amazon Cognito, scegli Pool di	Amministratore AWS

Operazione	Description	Competenze richieste
	utenti, quindi scegli il pool corrispondente `UserPoolId` (dagli CloudFormation output). 2. Aggiungi utenti al pool. Puoi scegliere tra due opzioni: • In Amazon Cognito, crea utenti manualmente con indirizzi e-mail e password temporanee (per ulteriori informazioni, consulta la documentazione di Amazon Cognito) oppure • Abilita l'autoregistrazione nell'interfaccia utente impostando il file. `hideSignUp={false}` `my-app/src/auth/AuthWrapper.jsx`	

Operazione	Description	Competenze richieste
Fornisci l'accesso Quick Sight alla dashboard	Per consentire l'accesso agli elementi visivi di Quick Sight, concedi l'autorizzazione Viewer agli utenti autenticati: 1. Nella console Quick Sight, scegli Dashboard. 2. Seleziona la dashboard che desideri condividere scegliendone il nome. 3. Nell'angolo in alto a destra della pagina del pannello di controllo, scegli Condividi. 4. Nel pannello di condivisi one, scegli Condividi con utenti specifici. 5. Per aggiungere nuovi utenti, scegli Invita utenti o l'icona più (+). 6. Nel campo e-mail, inserisci l'indirizzo email completo del destinatario. (Questa e-mail deve corrispon dere esattamente al nome di accesso ad Amazon Cognito dell'utente) 7. Dal menu Autorizzazione accanto al campo e-mail, scegli Viewer. 8. Per inviare l'invito e concedere l'accesso, scegli Condividi.	Amministratore Quick Sight

Operazione	Description	Competenze richieste
	Ogni utente riceverà un'email con un link alla dashboard. Puoi modificare le autorizza zioni in qualsiasi momento tramite il menu Condividi. Per ulteriori informazioni, consulta Concedere a singoli utenti e gruppi Amazon Quick Sight l'accesso a un pannello di controllo in Amazon Quick Sight nella documentazione di Amazon Quick.	

Crea e distribuisci il frontend React

Operazione	Description	Competenze richieste

Installa le dipendenze e crea il progetto.	Nella directory dell'appl icazione React, esegui i seguenti comandi per generare file di produzione ottimizzati:
	
	`cd my-app`
	`npm install`
	`npm run build`
		Sviluppatore di app

Carica i file di build su Amazon S3.	Carica tutti i file dalla `my-app/dist/` directory nel bucket S3 fornito da CloudForm ation (non caricare la cartella stessa).	Sviluppatore di app

Operazione	Description	Competenze richieste
Crea un'invalidazione. CloudFront	Sulla CloudFront console, crea un'invalidazione per il percorso di aggiornamento dei contenuti memorizzati nella cache `/*` dopo la distribuzione. Per istruzioni, consulta Invalidare i file nella documentazione. CloudFront	Amministratore AWS

Configura la lista delle autorizzazioni Quick Sight

Operazione	Description	Competenze richieste
Aggiungi il CloudFront dominio all'elenco delle autorizzazioni di Quick Sight.	Per consentire al tuo CloudFront dominio di incorporare in modo sicuro gli elementi visivi Quick Sight: 1. Sulla console Quick Sight, scegli Gestisci Quick Sight, Domains and Embedding. 2. Aggiungi il CloudFront dominio alla casella Dominio (ad esempio,`https://d1234567890abc.cloudfront.net`). 3. Scegli Add domain (Aggiungi dominio).	Amministratore Quick Sight

Accedi all'applicazione e verifica la funzionalità

Operazione	Description	Competenze richieste
Apri l'applicazione React.	Usa il CloudFront dominio (dagli CloudFormation output) per aprire l'applicazione web React distribuita in un browser.	Proprietario dell'app
Verifica l'autenticazione.	Accedi all'applicazione utilizzando le credenziali di Amazon Cognito per verificare il flusso di autenticazione e la convalida JWT tramite API Gateway.	Proprietario dell'app
Verifica gli elementi visivi incorporati.	Verifica che le immagini Quick Sight vengano caricate correttamente all'interno dell'applicazione in base alle autorizzazioni di accesso specifiche dell'utente.	Proprietario dell'app
Convalida la connettività API e Lambda.	Verifica che l'applicazione sia in grado di chiamare correttamente l'`/get-embed-url` API e recuperare un URLs incorporamento Quick Sight valido senza errori.	Proprietario dell'app

Monitora e gestisci l'applicazione

Operazione	Description	Competenze richieste
Monitora utilizzando CloudWatch.	È possibile utilizzare strumenti AWS di osservabilità per	Amministratore AWS

Operazione	Description	Competenze richieste
	monitorare l'applicazione e mantenere prestazioni sicure e scalabili in produzione. Esamina i log Lambda, le metriche di API Gateway e gli eventi di autenticazione di Amazon Cognito CloudWatch per garantire lo stato delle applicazioni e rilevare anomalie.	
Monitora AWS WAF e CloudFront registra.	Controlla AWS WAF i log per verificare la presenza di richieste bloccate o sospette e i log di CloudFront accesso per verificare le prestazioni e le metriche di memorizzazione nella cache.	Amministratore AWS

Risoluzione dei problemi

Problema	Soluzione
Errore «Dominio non consentito»	1. Nella console Quick Sight, scegli Gestisci Quick Sight, Domains and Embedding. 2. Aggiungi il tuo CloudFront dominio (ad esempio,`https://d1234567890abc.cloudfront.net`). 3. Salva la configurazione e ricarica l'applicazione.
Errori di autenticazione	Possibili cause:

Problema	Soluzione
	• ID del pool di utenti Amazon Cognito o ID client dell'applicazione errato nel `.env` file. • L'utente non è stato confermato nel pool di utenti di Amazon Cognito. • Token JWT non valido o mancante nelle richieste API. Soluzioni: • Verifica i valori di configurazione di Amazon Cognito nel `.env` file. • Sulla console Amazon Cognito, controlla la scheda Utenti per verificare che l'utente sia attivo. • Verifica che l'intestazione di autorizzazione contenga un token JWT valido. • Sulla console API Gateway, seleziona Authorizers per verificare che API Gateway utilizzi l'autorizzatore Amazon Cognito per l'autenticazione.

Problema	Soluzione
Errori dell'API Gateway	Possibili cause: • Autorizzatore Amazon Cognito mancante o configurato in modo errato in API Gateway. • Autorizzazioni Lambda insufficienti per chiamare l'API Quick Sight. • Mancata corrispondenza dell'URL dell'API tra `.env` gli output e gli output. CloudFormation Soluzioni: • Riconferma che l'URL di richiamo dell'API Gateway sia presente nel file `.env`. • Sulla console API Gateway, seleziona Authorizers per verificare la configurazione di Amazon Cognito. • Allega la policy IAM richiesta con `quicksight:GenerateEmbedUrl ForRegisteredUser` al ruolo di esecuzione Lambda. • CloudWatch Esamina i log per verificare eventuali errori dettagliati relativi all'API o alle autorizzazioni. Per ulteriori informazioni, consulta la documentazione di API Gateway.

Problema	Soluzione
Le immagini di Quick Sight non si caricano	Possibili cause: • ID dashboard Quick Sight, ID del foglio o ID visivo non corretti nella configurazione dell'ambiente. • Nessun accesso da parte dell'utente Quick Sight alla visualizzazione o al pannello di controllo. • Configurazione CORS (Cross-Origin Resource Sharing) mancante in API Gateway. Soluzioni: • Conferma Quick Sight IDs dal menu a tre punti nell'angolo in alto a destra della visualizzazione del pannello di controllo. • Verifica che l'utente Quick Sight abbia il ruolo di visualizzatore per la dashboard condivisa. • Abilita CORS in API Gateway e ridistribuisci l'API. Per ulteriori informazioni, consulta la documentazione di API Gateway. • Controllate la presenza di CORS nella console del browser per individuare eventuali errori relativi alla rete.

Problema	Soluzione
Errore «L'utente non ha accesso»	Possibili cause: • L'e-mail dell'utente non è stata condivisa con la dashboard di Quick Sight. • Ruolo o autorizzazione utente errati in Quick Sight. Soluzione:: 1. Sulla console Quick Sight, apri la dashboard, quindi scegli Condividi, Condividi dashboard. 2. Aggiungi l'indirizzo e-mail dell'utente (corrispondente all'e-mail di accesso ad Amazon Cognito). 3. Assegna il ruolo di autorizzazione Viewer. 4. Chiedi all'utente di disconnettersi e accedere nuovamente per aggiornare le autorizzazioni.

Risorse correlate

AWS documentazione

- Iscrizione a un abbonamento Amazon Quick
- Analisi integrate per Amazon Quick Sight
- Riferimento all'API Quick Sight: GenerateEmbedUrlForRegisteredUser
- Pool di utenti Amazon Cognito
- AWS Lambda Guida per gli sviluppatori
- Guida per sviluppatori di Amazon API Gateway
- Monitoraggio di base e monitoraggio dettagliato in CloudWatch
- AWS CloudFormation Guida per l'utente
- Guida per CloudFront sviluppatori Amazon
- AWS WAF Guida per gli sviluppatori

Tutorial e video

- Analisi integrate per Amazon Quick Sight
- YouTube Tutorial Amazon Quick Sight

Esplora lo sviluppo completo di applicazioni web native per il cloud con Green Boost

Ben Stickley e Amiin Samatar, Amazon Web Services

Riepilogo

In risposta alle esigenze in continua evoluzione degli sviluppatori, Amazon Web Services (AWS) riconosce la necessità fondamentale di un approccio efficiente allo sviluppo di applicazioni Web native per il cloud. L'obiettivo di AWS è aiutarti a superare gli ostacoli comuni associati alla distribuzione di app Web sul cloud AWS. Sfruttando le funzionalità di tecnologie moderne come TypeScript AWS Cloud Development Kit (AWS CDK), React e Node.js, questo modello mira a semplificare e accelerare il processo di sviluppo.

Sostenuto dal toolkit Green Boost (GB), il modello offre una guida pratica alla creazione di applicazioni Web che sfruttano appieno le ampie funzionalità di AWS. Funziona come una tabella di marcia completa, che ti guida attraverso il processo di implementazione di un'applicazione web CRUD (Create, Read, Update, Delete) fondamentale integrata con Amazon Aurora PostgreSQL Compatible Edition. Ciò si ottiene utilizzando l'interfaccia a riga di comando Green Boost (Green Boost CLI) e stabilendo un ambiente di sviluppo locale.

Dopo la corretta implementazione dell'applicazione, il modello approfondisce i componenti chiave dell'app Web, tra cui la progettazione dell'infrastruttura, lo sviluppo di backend e frontend e strumenti essenziali come cdk-dia per la visualizzazione, che facilitano una gestione efficiente del progetto.

Prerequisiti e limitazioni

Prerequisiti

- Git installato
- Visual Studio Code (VS Code) installato
- AWS Command Line Interface (AWS CLI) installata
- AWS CDK Toolkit installato
- Node.js 18 installato o Node.js 18 con pnpm attivato
- pnpm installato, se non fa parte dell'installazione di Node.js
- Familiarità di base con TypeScript AWS CDK, Node.js e React
- Un account AWS attivo

- Un account AWS avviato utilizzando AWS CDK in. `us-east-1` La regione `us-east-1` AWS è necessaria per il supporto delle funzioni Amazon CloudFront Lambda @Edge.
- Credenziali di sicurezza AWS`AWS_ACCESS_KEY_ID`, incluse quelle configurate correttamente nell'ambiente terminale
- Per gli utenti Windows, un terminale in modalità amministratore (per adattarsi al modo in cui pnpm gestisce i moduli dei nodi)

Versioni del prodotto

- SDK AWS per la JavaScript versione 3
- AWS CDK versione 2
- AWS CLI versione 2.2
- Node.js versione 18
- React versione 18

Architecture

Stack tecnologico Target

- Amazon Aurora PostgreSQL-Compatible Edition
- Amazon CloudFront
- Amazon CloudWatch
- Amazon Elastic Compute Cloud (Amazon EC2)
- AWS Lambda
- AWS Secrets Manager
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Storage Service (Amazon S3)
- AWS WAF

Architettura Target

Il diagramma seguente mostra che le richieste degli utenti passano attraverso Amazon CloudFront, AWS WAF e AWS Lambda prima di interagire con un bucket S3, un database Aurora, un'istanza

e infine raggiungere gli sviluppatori. EC2 Gli amministratori, invece, utilizzano Amazon SNS e CloudWatch Amazon per scopi di notifica e monitoraggio.

Per dare un'occhiata più approfondita all'applicazione dopo la distribuzione, puoi creare un diagramma utilizzando cdk-dia, come mostrato nell'esempio seguente.

Questi diagrammi mostrano l'architettura dell'applicazione web da due angolazioni distinte. Il diagramma cdk-dia offre una visione tecnica dettagliata dell'infrastruttura CDK di AWS, evidenziando servizi AWS specifici come la compatibilità con Amazon Aurora PostgreSQL e AWS Lambda. Al contrario, l'altro diagramma assume una prospettiva più ampia, enfatizzando il flusso logico dei dati e le interazioni degli utenti. La differenza principale sta nel livello di dettaglio: il cdk-dia approfondisce le complessità tecniche, mentre il primo diagramma offre una visione più incentrata sull'utente.

La creazione del diagramma cdk-dia è trattata nell'epico Understand the app infrastructure by AWS CDK.

Tools (Strumenti)

Servizi AWS

- Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL.
- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software che aiuta a definire e fornire l'infrastruttura cloud AWS in codice.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.
- Amazon CloudFront accelera la distribuzione dei tuoi contenuti web distribuendoli attraverso una rete mondiale di data center, che riduce la latenza e migliora le prestazioni.
- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue risorse AWS e delle applicazioni che esegui su AWS in tempo reale.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.

- AWS Secrets Manager ti aiuta a sostituire le credenziali codificate nel codice, comprese le password, con una chiamata API a Secrets Manager per recuperare il segreto a livello di codice.
- AWS Systems Manager ti aiuta a gestire le applicazioni e l'infrastruttura in esecuzione nel cloud AWS. Semplifica la gestione delle applicazioni e delle risorse, riduce i tempi di rilevamento e risoluzione dei problemi operativi e ti aiuta a gestire le tue risorse AWS in modo sicuro su larga scala. Questo modello utilizza AWS Systems Manager Session Manager.
- Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è un servizio di storage di oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati. Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.
- AWS WAF è un firewall per applicazioni Web che ti aiuta a monitorare le richieste HTTP e HTTPS che vengono inoltrate alle risorse delle tue applicazioni Web protette.

Altri strumenti

- Git è un sistema di controllo delle versioni distribuito e open source.
- Green Boost è un toolkit per la creazione di app Web su AWS.
- Next.js è un framework React per aggiungere funzionalità e ottimizzazioni.
- Node.js è un ambiente di JavaScript runtime basato sugli eventi progettato per la creazione di applicazioni di rete scalabili.
- pgAdmin è uno strumento di gestione open source per PostgreSQL. Fornisce un'interfaccia grafica che consente di creare, gestire e utilizzare oggetti di database.
- pnpm è un gestore di pacchetti per le dipendenze del progetto Node.js.

Best practice

Consulta la sezione Epics per ulteriori informazioni sui seguenti consigli:

- Monitora l'infrastruttura utilizzando i CloudWatch pannelli di controllo e gli allarmi di Amazon.
- Applica le best practice di AWS utilizzando cdk-nag per eseguire analisi statiche dell'infrastruttura come codice (IaC).
- Stabilisci l'inoltro delle porte DB tramite il tunneling SSH (Secure Shell) con Systems Manager Session Manager, che è più sicuro rispetto all'avere un indirizzo IP esposto pubblicamente.
- Gestisci le vulnerabilità eseguendo. `pnpm audit`

- Applica le migliori pratiche utilizzando ESLintper eseguire analisi statiche TypeScript del codice e Prettier per standardizzare la formattazione del codice.

Epiche

Implementa un'app web CRUD con Aurora compatibile con PostgreSQL

Operazione	Description	Competenze richieste
Installa la CLI Green Boost.	Per installare Green Boost CLI, esegui il seguente comando. `pnpm add -g gboost`	Sviluppatore di app
Crea un'app GB.	1. Per creare un'app utilizzando Green Boost, esegui il comando`gboost create`. 2. Scegli il `CRUD App with Aurora PostgreSQL` modello.	Sviluppatore di app
Installa le dipendenze e distribuisci l'app.	1. Vai alla directory del progetto:. `cd <your directory>` 2. Per installare le dipendenze, esegui il comando`pnpm i`. 3. Vai alla directory infra:. `cd infra` 4. Per distribuire l'app localmente, esegui il comando. `pnpm deploy:local`	Sviluppatore di app

Operazione	Description	Competenze richieste
	Si tratta di un alias per un `cdk deploy ...` comando definito in. `infra/package.json` Attendi il termine della distribuzione (circa 20 minuti). Durante l'attesa, monitora gli CloudFormation stack AWS nella CloudFormation console. Nota come i costrutti definiti nel codice si associano alla risorsa distribuita. Esamina la visualizzazione ad albero di CDK Construct nella console. CloudFormation	

Operazione	Description	Competenze richieste
Accedi all'app.	Dopo aver distribuito l'app GB localmente, puoi accedervi utilizzando l' CloudFront URL. L'URL è stampato nell'output del terminale, ma può essere un po' difficile da trovare. Per trovarlo più rapidamente, segui i seguenti passaggi: 1. Apri il terminale in cui hai eseguito il `pnpm deploy:local` comando. 2. Cerca una sezione nell'output del terminale che assomigli al testo seguente. `myapp5stickbui9C39A55A.CloudFrontDomainName = d1q16n5pof924c.cloudfront.net` L'URL sarà unico per la tua distribuzione. In alternativa, puoi trovare l' CloudFront URL accedendo alla CloudFront console Amazon: 1. Accedi alla Console di gestione AWS e accedi al CloudFront servizio. 2. Cerca l'ultima distribuzione distribuita nell'elenco.	Sviluppatore di app

Operazione	Description	Competenze richieste
	Copia il nome di dominio associato alla distribuzione. Assomiglierà a `your-unique-id.cloudfront.net` .	

Monitora utilizzando Amazon CloudWatch

Operazione	Description	Competenze richieste
Visualizza la CloudWatch dashboard.	1. Apri la CloudWatch console e scegli Dashboard. 2. Seleziona la dashboard con il nome - -dashboard <appId><stageName>. 3. Controlla la dashboard . Quali risorse vengono monitorate? Quali metriche vengono registrate? Questa dashboard è resa possibile dal costrutto open source. cdk-monitoring-constructs	Sviluppatore di app
Abilita gli avvisi.	Una CloudWatch dashboard ti aiuta a monitorare attivamente la tua app web. Per monitorare passivamente la tua app web, puoi abilitare gli avvisi. 1. Vai a `/infra/src/app/stateless/monitor-stack.ts` , che definisce lo stack di monitor.	Sviluppatore di app

Operazione	Description	Competenze richieste

	2. Decommenta la riga seguente e sostituiscila `admin@example.com` con il tuo indirizzo email.
	`onAlarmTopic.addSubscription(new EmailSubscription("admin@example.com"));`
	3. Aggiungi le seguenti informazioni di importazione nella parte superiore del file.
	`import { EmailSubscription } from "aws-cdk-lib/aws-sns-subscriptions";`
	4. All'interno`infra/`, esegui il comando seguente.
	`cdk deploy "*/monitor" --exclusively.`
	5. Per confermare la tua iscrizione all'argomento SNS che viene avviato quando viene avviato un allarme di monitoraggio, scegli il link nel messaggio e-mail.	

Comprendi l'infrastruttura delle app utilizzando AWS CDK

Operazione	Description	Competenze richieste
Crea un diagramma di architettura.	Genera un diagramma di architettura della tua app web usando cdk-dia. La visualizzazione dell'architettura aiuta a migliorare la comprensione e la comunicazione tra i membri del team. Fornisce una panoramica chiara dei componenti del sistema e delle loro relazioni. 1. Installa Graphviz. 2. All'interno `infra/`, esegui il comando. `pnpm cdk-dia` 3. Visualizza il tuo `infra/diagram.png` .	Sviluppatore di app
Usa cdk-nag per applicare le migliori pratiche.	Usa cdk-nag per aiutarti a mantenere un'infrastruttura sicura e conforme applicando le migliori pratiche e riducendo il rischio di vulnerabilità di sicurezza e configurazioni errate. 1. Esplora l'applicazione delle best practice di cdk-nag nella sua sezione sulle regole, inclusi i controlli del Rules Pack della AWS Solutions Library. 2. Per vedere come cdk-nag applica le regole, apporta	Sviluppatore di app

Operazione	Description	Competenze richieste
	una modifica al codice. Ad esempio, in, cambia in`infra/src/app/stateful/data-stacks.ts` . `storageEncrypted: true` `storageEncrypted: false` 3. All'interno`infra/`, esegui il comando`cdk synth "*/data"`. Durante la sintesi, si verificherà un errore di compilazione che indica una violazione della regola. `AwsSolutions-RDS2: The RDS instance or Aurora DB cluster does not have storage encryption enabled.` Questo errore mostra come cdk-nag sia un meccanismo di sicurezza per far rispettare le migliori pratiche dell'infrastruttura e prevenire configurazioni errate di sicurezza. 4. Se necessario, puoi anche sopprimere le regole in ambiti diversi. Ad esempio, per sopprimere AwsSolutions -RDS2, aggiungete il codice seguente	

Operazione	Description	Competenze richieste

	sotto l'istanziazione di. `DbIamCluster`
	
	`NagSuppressions.addResourceSuppressions(`
	`  cluster.node.findChild("Resource"),`
	`  [`
	`  {`
	`  id: "AwsSolutions-RDS2",`
	`  reason: "Customer requirement necessitates having unencrypted DB storage",`
	`  },`
	`  ],`
	`);`
	
	5. Dopo la soppressione, esegui di nuovo. `cdk synth "*/data"` La tua app AWS CDK dovrebbe ora essere sintetizzata correttamente. Puoi trovare tutte le regole soppresse in. `infra/cdk.out/assembly-<appId>-<stageName>/AwsSolutions-<appId>-<stageName>-${stackId}-NagReport.csv`	

Valuta la configurazione e lo schema del database

Operazione	Description	Competenze richieste
Acquisire variabili di ambiente.	Per ottenere le variabili di ambiente richieste, utilizzate i seguenti passaggi: 1. Per trovarle`DB_BASTION_ID`, accedi alla console e accedi alla EC2 console. Scegli Istanze (in esecuzione) e trova la riga che contiene - ssm-db-bastion Nome<stageName>. L'ID dell'istanza inizia con i-. 2. Per trovarla**`DB_ENDPOINT`**, sulla console Amazon Relational Database Service (Amazon RDS), scegli Istanze database e seleziona il cluster regionale con un identificatore DB che inizia con - -data-. <appId><stageName> Individua l'endpoint dell'istanza writer, che termina con rds.amazonaws.com.	Sviluppatore di app
Stabilisci il port forwarding.	Per stabilire il port forwarding, utilizzare i seguenti passaggi: 1. Installa il plug-in AWS Systems Manager Session Manager.	Sviluppatore di app

Operazione	Description	Competenze richieste
	2. Inizia il port forwardin g eseguendo `pnpm db:connect` within `core/` per stabilire una connessione sicura tramite l'host bastion. 3. Dopo aver visualizzato il testo `Waiting for connections...,` nel terminale, è stato stabilito con successo un tunnel SSH tra la macchina locale e il server Aurora tramite EC2 l'host bastion.	
Regola il timeout di Systems Manager Session Manager.	(Facoltativo) Se il timeout di sessione predefinito di 20 minuti è troppo breve, è possibile aumentarlo fino a 60 minuti nella console Systems Manager selezionando Gestione sessioni, Preferenz e, Modifica, Timeout sessione inattiva.	Sviluppatore di app

Operazione	Description	Competenze richieste
Visualizza il database.	pgAdmin è uno strumento open source intuitivo per la gestione dei database PostgreSQL. Semplifica le attività del database, consentendoti di creare, gestire e ottimizzare i database in modo efficient e. Questa sezione guida l'utente nell'installazione di pgAdmin e nell'utilizzo delle sue funzionalità per la gestione del database PostgreSQL. 1. In Object Explorer, apri il menu contestuale (fai clic con il pulsante destro del mouse) per Server, quindi scegli Registra, Server. 2. Nella scheda Generale, inserisci - <appId><s tageName>per il campo Nome. 3. Per recuperare la password del DB, apri la console AWS Secrets Manager, seleziona il segreto con la descrizione Generato dal CDK per lo stack: - - data e scegli la scheda Secret Value. <appId><s tageName> Scegli Recupera valore segreto e	Sviluppatore di app

Operazione	Description	Competenze richieste
	copia il valore segreto con una chiave di password. 4. Nella scheda Connessione, immettere 0.0.0 per il name/address campo Host e immettere _admin per il campo Nome utente. <appId> Per il campo Password, usa il segreto che hai recuperato in precedenza. Scegli sì per salvare la password? campo. 5. Scegli Save (Salva). 6. Per visualizzare le tabelle, accedi a -, Databases, _db, Schemas, Tables. <appId><stageName> <appId><appId> 7. Apri il menu contestuale (fai clic con il pulsante destro del mouse) per la tabella degli elementi, quindi seleziona Visualizza/Modifica dati, tutte le righe. 8. Esplora la tabella.	

Esegui il debug con Node.js

Operazione	Description	Competenze richieste

Esegui il debug dello use case create item.	Per eseguire il debug del caso d'uso di creazione dell'elem ento, segui questi passaggi:
	
	1. Apri il `core/src/modules/item/create-item.use-case.ts` file e inserisci il codice seguente.
	
	``
	import { fileURLTo
	Path } from "node:url
	";
	
	// existing create-it
	em.use-case.ts code
	 here
	
	if (process.argv[1]
	 === fileURLTo
	Path(import.meta.u
	rl)) {
	  createItemUseCase(
	{
	    description:
	 "Item 1's Descripti
	on",
	    name: "Item 1",
	  });
	}
	``
	
	2. Il codice aggiunto nel passaggio precedente assicura che la `createIte mUseCase` funzione venga chiamata quando questo modulo viene eseguito
		Sviluppatore di app

Operazione	Description	Competenze richieste
	direttamente. Imposta i punti di interruzione sulle righe all'interno di questo blocco di codice in cui desideri avviare line-by-line il debug. 1. Apri il terminale di JavaScript debug VS Code, quindi esegui `pnpm tsx core/src/modules/item/create-item.use-case.ts` per eseguire il codice con il debug. line-by-line In alternativa, puoi utilizzare `console.log` le istruzioni, ma le istruzioni stampate possono essere inadeguate quando lavori con logiche aziendali complesse. Line-by-lineil debug offre un contesto più ampio.	

Sviluppa il frontend

Operazione	Description	Competenze richieste
Configura il server di sviluppo.	1. Accedere al `ui/` server di sviluppo Next.js ed eseguirlo `pnpm dev` per avviarlo. 2. Accedi alla tua app web localmente all'indirizzo	Sviluppatore di app

Operazione	Description	Competenze richieste
	`http://localhost:3000` . Il server di sviluppo Next.js è configurato con il feedback istantaneo Fast Refresh sulle modifiche apportate ai componenti React. 3. Sperimenta con la personalizzazione del colore della barra dell'app. Apri il `ui/src/components/theme/theme.tsx` file e individua la sezione che definisce il tema per la barra dell'app. Nella `colorSchemes.light.palette.primary` sezione, aggiorna il valore principale da `colors.lagoon` a `colors.carrot` . Dopo aver apportato questa modifica, salva il file e osserva l'aggiornamento nel tuo browser. 4. Sperimenta modificando testo, componenti e aggiungendo nuove pagine.	

Utensili con Green Boost

Operazione	Description	Competenze richieste
Configura monorepo e il gestore di pacchetti pnpm.	1. `pnpm-workspace.yaml` Esamina nella radice del tuo repository GB e nota come vengono definiti gli spazi di lavoro. Per ulteriori informazioni sugli spazi di lavoro, consulta la documentazione di pnpm. 2. `ui/package.json` Esamina e nota come fa riferimento allo spazio di lavoro `core/` con il nome del pacchetto . `"<appId>/core": "workspace:^",` 3. Osserva come TypeScript la ESLint configurazione è centralizzata nei pacchetti di utilità definiti all'interno. `packages/` Questa configurazione viene quindi utilizzata da pacchetti applicativi come `core/infra/`, `eui/`. Ciò è utile quando l'app è ridimensionata e si definiscono più pacchetti applicativi, che possono fare riferimento ai pacchetti di utilità senza duplicare il codice di configurazione.	Sviluppatore di app

Operazione	Description	Competenze richieste
Esegui script pnpm.	Esegui i seguenti comandi nella directory principale del tuo repository: 1. Esegui `pnpm lint`. Questo comando esegue l'analisi statica del codice con ESLint. 2. Esegui `pnpm typecheck`. Questo comando esegue il TypeScript compilatore per controllare i tipi di codice. 3. Esegui `pnpm test`. Questo comando esegue Vitest per eseguire test unitari. Notate come questi comandi vengono eseguiti in tutte le aree di lavoro. I comandi sono definiti nel campo di ogni area di lavoro di `package.json#scripts`.	Sviluppatore di app

Operazione	Description	Competenze richieste
Utilizzare ESLint per l'analisi statica del codice.	Per testare la capacità di analisi statica del codice ESLint, effettuate le seguenti operazioni: 1. Innanzitutto, assicurati che l' ESLint estensione VS Code (ID:`dbaeumer.vscode-eslint`) sia installata. Ti consigliamo di installare anche VS Code Error Lens (ID:`usernamehw.errorlens`) per visualizzare gli errori in linea. 2. Nel codice, includete intenzionalmente una riga di codice che utilizza la `eval()` funzione, come illustrato nell'esempio seguente. `const userInput = "import("fs").then((fs) => console.log(fs.readFileSync("/etc/passwd", { encoding: "utf8" })))"; eval(userInput);` **Important** Questo è solo a scopo di test. L'uso `eval()`	Sviluppatore di app

Operazione	Description	Competenze richieste
	è considerato potenzialmente pericoloso e deve essere evitato a causa dei rischi per la sicurezza. 3. Dopo aver incluso la `eval()` riga, apri l'editor di codice per confermare che l'odore del codice è ESLint indicato utilizzando degli scarabocchi rossi. 4. Controlla i ESLint plugin e la configurazione su. `packages/eslint-config-{node,next}/.eslintrc.cjs`	

Operazione	Description	Competenze richieste
Gestisci dipendenze e vulnerabilità.	1. Per identificare eventuali vulnerabilità ed esposizioni comuni (CVEs), esegui `pnpm audit` nella radice del tuo repository. Dovresti vedere Nessuna vulnerabilità nota trovata. 2. Installa un pacchetto intenzionalmente vulnerabile all'interno `core/` eseguendolo`pnpm add minimist@0.2.3` , quindi eseguilo. `pnpm audit` Notate la vulnerabilità segnalata. 3. Disinstalla il pacchetto vulnerabile all'interno `core/` **`pnpm remove minimist`**eseguendo.	Sviluppatore di app

Operazione	Description	Competenze richieste
Hook di pre-commit con Husky.	1. Apporta un paio di piccole modifiche ai TypeScript file in tutto il repository. Le modifiche possono essere semplici come l'aggiunta di commenti. 2. Pianifica e conferma queste modifiche utilizzando `git add -A` e then`git commit -m "test husky"`. Il trigger Husky pre-commi t hook, definito in `.husky/pre-commit` , esegue il comando. `pnpm lint-staged` 3. Osserva come lint-stag ed esegue i comandi specificati nei `*/.lintstagedrc.js` file in tutto il repository su file che sono stati gestiti da Git. Questi strumenti sono meccanismi che aiutano a impedire che codice errato entri nell'applicazione.	Sviluppatore di app

Distruggi l'infrastruttura

Operazione	Description	Competenze richieste
Rimuovi la distribuzione dal tuo account.	1. Per demolire l'infrastruttura che hai fornito nella prima epopea, tramite run in. `pnpm destroy:local infra/` 2. Attendi 15 minuti dopo `pnpm destroy:local` il completamento, quindi elimina la funzione Lambda @Edge mantenuta cercando l'ID dell'app nella console Lambda. Le funzioni Lambda @Edge vengono replicate., il che le rende difficili da eliminare . Per ulteriori informazioni sull'eliminazione delle funzioni Lambda @Edge, consulta CloudFront la documentazione.	Sviluppatore di app

risoluzione dei problemi

Problema	Soluzione
Impossibile stabilire il port forwarding	Assicurati che le tue credenziali AWS siano configurate correttamente e dispongano delle autorizzazioni necessarie. Ricontrolla che le variabili di ambiente bastion host ID (`DB_BASTION_ID`) e database

Problema	Soluzione
	endpoint (`DB_ENDPOINT`) siano impostate correttamente. Se i problemi persistono, consulta la documentazione AWS per la risoluzione dei problemi di connessioni SSH e Session Manager.
Il sito Web non si sta caricando `localhost:3000`	Verifica che l'output del terminale indichi che il port forwarding è andato a buon fine, incluso l'indirizzo di inoltro. Assicurati che non vi siano processi in conflitto utilizzando la porta 3000 sul computer locale. Verificate che l'applicazione Green Boost sia configurata correttamente e in esecuzione sulla porta prevista (3000). Controlla il tuo browser web per eventuali estensioni o impostazioni di sicurezza che potrebbero bloccare le connessioni locali.
Messaggi di errore durante la distribuzione locale (`pnpm deploy:local`)	Esamina attentamente i messaggi di errore per identificare la causa del problema. Verificate che le variabili di ambiente e i file di configurazione necessari siano impostati correttamente.

Risorse correlate

- Documentazione CDK AWS
- Documentazione Green Boost
- Documentazione Next.js
- Documentazione Node.js

- Documentazione React
- TypeScript documentazione

Esegui test unitari per un'applicazione Node.js GitHub utilizzando AWS CodeBuild

Thomas Scott e Jean-Baptiste Guillois, Amazon Web Services

Riepilogo

Questo modello fornisce codice sorgente di esempio e componenti di test unitari chiave per un'API di gioco Node.js. Include anche istruzioni per eseguire questi unit test da un GitHub repository utilizzando AWS CodeBuild, come parte del flusso di lavoro di integrazione continua e distribuzione continua (CI/CD).

Lo unit test è un processo di sviluppo software in cui diverse parti di un'applicazione, chiamate unità, vengono testate individualmente e indipendentemente per verificarne il corretto funzionamento. I test convalidano la qualità del codice e confermano che funzioni come previsto. Anche altri sviluppatori possono facilmente acquisire familiarità con la vostra base di codice consultando i test. I test unitari riducono i tempi di refactoring futuri, aiutano gli ingegneri ad aggiornarsi più rapidamente sulla base di codice e forniscono fiducia nel comportamento previsto.

Il test unitario prevede il test di singole funzioni, incluse le funzioni AWS Lambda. Per creare test unitari, è necessario un framework di test e un modo per convalidare i test (asserzioni). Gli esempi di codice in questo modello utilizzano il framework di test Mocha e la libreria di asserzioni Chai.

Per ulteriori informazioni sui test unitari ed esempi di componenti di test, consultate la sezione Informazioni aggiuntive.

Prerequisiti e limitazioni

- Un account AWS attivo con CodeBuild autorizzazioni corrette
- Un GitHub account (consulta le istruzioni per la registrazione)
- Git (vedi istruzioni di installazione)
- Un editor di codice per apportare modifiche e inviare il codice a GitHub

Architecture

Questo modello implementa l'architettura mostrata nel diagramma seguente.

Tools (Strumenti)

Strumenti

- Git è un sistema di controllo delle versioni che puoi usare per lo sviluppo del codice.
- AWS CodeBuild è un servizio di integrazione continua completamente gestito che compila codice sorgente, esegue test e produce pacchetti software pronti per la distribuzione. Con CodeBuild, non è necessario fornire, gestire e scalare i propri server di build. CodeBuild esegue la scalabilità continua ed elabora più build contemporaneamente, in modo che le build non restino in attesa in coda. Puoi iniziare a utilizzare CodeBuild velocemente con ambienti di compilazione predefiniti oppure puoi creare ambienti di compilazione personalizzati che utilizzano strumenti di compilazione specifici. Con CodeBuild, ti vengono addebitati al minuto per le risorse di calcolo che utilizzi.

Codice

Il codice sorgente di questo pattern è disponibile su GitHub, nel repository dell'applicazione Sample game unit test. È possibile creare il proprio GitHub repository da questo esempio (opzione 1) o utilizzare direttamente il repository di esempio (opzione 2) per questo modello. Segui le istruzioni per ciascuna opzione nella sezione successiva. L'opzione che segui dipenderà dal tuo caso d'uso.

Epiche

Opzione 1: esegui test unitari sul tuo GitHub repository personale con CodeBuild

Operazione	Description	Competenze richieste
Crea il tuo GitHub repository sulla base del progetto di esempio.	1. Effettua il login a. GitHub 2. Crea un nuovo repository. Per istruzioni, consulta la GitHub documentazione. 3. Clona e trasferisci il repository di esempio nel nuovo repository del tuo account.	Sviluppatore di app, amministratore AWS, AWS DevOps
Crea un nuovo CodeBuild progetto.	1. Accedi alla Console di gestione AWS e apri la CodeBuild console su	Sviluppatore di app, amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste
	https://console.aws.amazon.com/codesuite/codebuild/home. 2. Scegliere Create build project (Crea progetto di compilazione). 3. Nella sezione Configurazione del progetto, per Project name, digita -js. aws-tests-sample-node 4. Nella sezione Source, per Source provider, scegli GitHub. 5. Per Repository, scegli Repository nel mio GitHub account, quindi incolla l'URL nel repository appena creato GitHub . 6. Nella sezione Primary source webhook events, seleziona Ricostruisci ogni volta che viene inviata una modifica al codice in questo repository. 7. Per il tipo di evento, scegli PUSH. 8. Nella sezione Ambiente, scegli Immagine gestita, Amazon Linux e l'immagine più recente. 9. Lascia le impostazioni predefinite per tutte le altre	

Operazione	Description	Competenze richieste
	opzioni, quindi scegli Crea progetto di compilazione.	
Inizia la compilazione.	Nella pagina Review (Verifica), selezionare Start build (Avvia compilazione) per eseguire la compilazione.	Sviluppatore di app, amministratore AWS, AWS DevOps

Opzione 2: esegui test unitari su un repository pubblico con CodeBuild

Operazione	Description	Competenze richieste
Crea un nuovo progetto di CodeBuild build.	1. Accedi alla Console di gestione AWS e apri la CodeBuild console su https://console.aws.amazon.com/codesuite/codebuild/home. 2. Scegliere Create build project (Crea progetto di compilazione). 3. Nella sezione Configurazione del progetto, per Project name, digita -js. aws-tests-sample-node 4. Nella sezione Source, per Source provider, scegli GitHub. 5. Per Repository, scegli Archivio pubblico, quindi incolla l'URL:. https://github.com/aws-samples/node-js-tests-sample	Sviluppatore di app, amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste
	6. Nella sezione Ambiente, scegli Immagine gestita, Amazon Linux e l'immagine più recente. 7. Lascia le impostazioni predefinite per tutte le altre opzioni, quindi scegli Crea progetto di compilazione.	
Inizia la compilazione.	Nella pagina Review (Verifica), selezionare Start build (Avvia compilazione) per eseguire la compilazione.	Sviluppatore di app, amministratore AWS, AWS DevOps

Analizza i test unitari

Operazione	Description	Competenze richieste
Visualizza i risultati dei test.	Nella CodeBuild console, esamina i risultati del test unitario del CodeBuild lavoro. Dovrebbero corrispondere ai risultati mostrati nella sezione Informazioni aggiuntive. Questi risultati convalidano l'integrazione del GitHub repository con. CodeBuild	Sviluppatore di app, amministratore AWS, AWS DevOps
Applica un webhook.	Ora puoi applicare un webhook, in modo da poter avviare automaticamente una build ogni volta che inserisci modifiche al codice nel ramo principale del tuo repositor	Sviluppatore di app, amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste
	y. Per istruzioni, consulta la documentazione. CodeBuild	

Risorse correlate

- Esempio di applicazione di game unit test (GitHub repository con codice di esempio)
- CodeBuild Documentazione AWS
- GitHub eventi webhook (CodeBuild documentazione)
- Creazione di un nuovo repository (documentazione) GitHub

Informazioni aggiuntive

Risultati dei test unitari

Nella CodeBuild console, dovresti vedere i seguenti risultati dei test dopo che il progetto è stato compilato correttamente.

Componenti di test unitari di esempio

Questa sezione descrive i quattro tipi di componenti di test utilizzati nei test unitari: asserzioni, spie, stub e mock. Include una breve spiegazione e un esempio di codice di ciascun componente.

Asserzioni

Un'asserzione viene utilizzata per verificare un risultato previsto. Questo è un componente di test importante perché convalida la risposta prevista da una determinata funzione. L'asserzione di esempio seguente verifica che l'ID restituito sia compreso tra 0 e 1000 quando si inizializza un nuovo gioco.

```
const { expect }  = require('chai');
const { Game } = require('../src/index');

describe('Game Function Group', () => {
 it('Check that the Game ID is between 0 and 1000', function() {
      const game = new Game();
      expect(game.id).is.above(0).but.below(1000)
```

```
 });
});
```

Spie

Una spia viene utilizzata per osservare cosa succede quando una funzione è in esecuzione. Ad esempio, potresti voler verificare che la funzione sia stata chiamata correttamente. L'esempio seguente mostra che i metodi start e stop vengono chiamati su un oggetto della classe Game.

```
const { expect }  = require('chai');
const { spy } = require('sinon');

const { Game } = require('../src/index');

describe('Game Function Group', () => {
   it('should verify that the correct function is called', () => {
      const spyStart = spy(Game.prototype, "start");
      const spyStop = spy(Game.prototype, "stop");

      const game = new Game();
      game.start();
      game.stop();

      expect(spyStart.called).to.be.true
      expect(spyStop.called).to.be.true
    });
});
```

Stub

Uno stub viene utilizzato per sovrascrivere la risposta predefinita di una funzione. Ciò è particolarmente utile quando la funzione effettua una richiesta esterna, perché si desidera evitare di effettuare richieste esterne dai test unitari. (Le richieste esterne sono più adatte per i test di integrazione, che possono testare fisicamente le richieste tra diversi componenti.) Nell'esempio seguente, uno stub impone un ID di ritorno dalla funzione getID.

```
const { expect }  = require('chai');
const {.stub } = require('sinon');

const { Game } = require('../src/index');

describe('Game Function Group', () =>  {
```

```
   it('Check that the Game ID is between 0 and 1000', function() {
      let generateIdStub = stub(Game.prototype, 'getId').returns(999999);

      const game = new Game();

      expect(game.getId).is.equal(999999);

      generateIdStub.restore();
    });
});
```

Simulazioni

Un simulato è un metodo falso che ha un comportamento preprogrammato per testare diversi scenari. Un mock può essere considerato una forma estesa di stub e può svolgere più attività contemporaneamente. Nell'esempio seguente, un mock viene utilizzato per convalidare tre scenari:

- La funzione viene chiamata
- La funzione viene chiamata con argomenti
- La funzione restituisce il numero intero 9

```
const { expect }  = require('chai');
const {.mock } = require('sinon');

const { Game } = require('../src/index');

describe('Game Function Group', () =>  {
   it('Check that the Game ID is between 0 and 1000', function() {
      let mock = mock(Game.prototype).expects('getId').withArgs().returns(9);

      const game = new Game();
      const id = get.getId();

      mock.verify();
      expect(id).is.equal(9);
    });
});
```

Struttura un progetto Python in architettura esagonale utilizzando AWS Lambda

Furkan Oruc, Dominik Goby, Darius Kunce e Michal Ploski, Amazon Web Services

Riepilogo

Questo modello mostra come strutturare un progetto Python in architettura esagonale utilizzando AWS Lambda. Il modello utilizza AWS Cloud Development Kit (AWS CDK) come strumento Infrastructure as code (IaC), Amazon API Gateway come API REST e Amazon DynamoDB come livello di persistenza. L'architettura esagonale segue i principi di progettazione basati sul dominio. Nell'architettura esagonale, il software è composto da tre componenti: dominio, porte e adattatori. Per informazioni dettagliate sulle architetture esagonali e sui relativi vantaggi, consulta la guida Building hexagonal architectures on AWS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Esperienza in Python
- Familiarità con AWS Lambda, AWS CDK, Amazon API Gateway e DynamoDB
- Un GitHub account (consulta le istruzioni per la registrazione)
- Git (vedi istruzioni di installazione)
- Un editor di codice per apportare modifiche e inviare codice GitHub (ad esempio, Visual Studio Code o JetBrains PyCharm)
- Docker installato e il demone Docker attivo e funzionante

Versioni del prodotto

- Git versione 2.24.3 o successiva
- Python versione 3.7 o successiva
- CDK AWS v2
- Poetry versione 1.1.13 o successiva
- AWS Lambda Powertools per Python versione 1.25.6 o successiva

- pytest versione 7.1.1 o successiva
- Moto versione 3.1.9 o successiva
- pydantic versione 1.9.0 o successiva
- Boto3 versione 1.22.4 o successiva
- mypy-boto3-dynamodb versione 1.24.0 o successiva

Architecture

Stack tecnologico Target

Lo stack tecnologico di destinazione è costituito da un servizio Python che utilizza API Gateway, Lambda e DynamoDB. Il servizio utilizza un adattatore DynamoDB per rendere persistenti i dati. Fornisce una funzione che utilizza Lambda come punto di ingresso. Il servizio utilizza Amazon API Gateway per esporre un'API REST. L'API utilizza AWS Identity and Access Management (IAM) per l'autenticazione dei client.

Architettura Target

Per illustrare l'implementazione, questo modello implementa un'architettura di destinazione senza server. I client possono inviare richieste a un endpoint API Gateway. API Gateway inoltra la richiesta alla funzione Lambda di destinazione che implementa il modello di architettura esagonale. La funzione Lambda esegue operazioni di creazione, lettura, aggiornamento ed eliminazione (CRUD) su una tabella DynamoDB.

> **Important**
>
> Questo modello è stato testato in un ambiente PoC. È necessario condurre una revisione della sicurezza per identificare il modello di minaccia e creare una base di codice sicura prima di implementare qualsiasi architettura in un ambiente di produzione.

L'API supporta cinque operazioni su un'entità di prodotto:

- `GET /products`restituisce tutti i prodotti.
- `POST /products`crea un nuovo prodotto.

- `GET /products/{id}`restituisce un prodotto specifico.
- `PUT /products/{id}`aggiorna un prodotto specifico.
- `DELETE /products/{id}`elimina un prodotto specifico.

È possibile utilizzare la seguente struttura di cartelle per organizzare il progetto in modo da seguire lo schema di architettura esagonale:

```
app/  # application code
|--- adapters/  # implementation of the ports defined in the domain
     |--- tests/  # adapter unit tests
|--- entrypoints/  # primary adapters, entry points
     |--- api/  # api entry point
          |--- model/  # api model
          |--- tests/  # end to end api tests
|--- domain/  # domain to implement business logic using hexagonal architecture
     |--- command_handlers/  # handlers used to execute commands on the domain
     |--- commands/  # commands on the domain
     |--- events/  # events triggered via the domain
     |--- exceptions/  # exceptions defined on the domain
     |--- model/  # domain model
     |--- ports/  # abstractions used for external communication
     |--- tests/  # domain tests
|--- libraries/  # List of 3rd party libraries used by the Lambda function
infra/  # infrastructure code
simple-crud-app.py  # AWS CDK v2 app
```

Tools (Strumenti)

Servizi AWS

- Amazon API Gateway è un servizio completamente gestito che semplifica per gli sviluppatori la creazione, la pubblicazione, la manutenzione, il monitoraggio e la protezione APIs su qualsiasi scala.
- Amazon DynamoDB è un database NoSQL chiave-valore completamente gestito, serverless e progettato per eseguire applicazioni ad alte prestazioni su qualsiasi scala.
- AWS Lambda è un servizio di elaborazione serverless e basato sugli eventi che consente di eseguire codice per praticamente qualsiasi tipo di applicazione o servizio di backend senza dover fornire o gestire server. Puoi lanciare funzioni Lambda da oltre 200 servizi AWS e applicazioni SaaS (Software as a Service) e pagare solo per ciò che usi.

Strumenti

- Git viene utilizzato come sistema di controllo della versione per lo sviluppo del codice in questo modello.
- Python è usato come linguaggio di programmazione per questo modello. Python fornisce strutture di dati di alto livello e un approccio alla programmazione orientata agli oggetti. AWS Lambda fornisce un runtime Python integrato che semplifica il funzionamento dei servizi Python.
- Visual Studio Code viene utilizzato come IDE per lo sviluppo e il test di questo modello. Puoi usare qualsiasi IDE che supporti lo sviluppo in Python (ad esempio, PyCharm).
- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software open source che consente di definire le risorse delle applicazioni cloud utilizzando linguaggi di programmazione familiari. Questo modello utilizza il CDK per scrivere e distribuire l'infrastruttura cloud come codice.
- La poesia viene utilizzata per gestire le dipendenze del modello.
- Docker viene utilizzato da AWS CDK per creare il pacchetto e il layer Lambda.

Codice

Il codice per questo pattern è disponibile nell'archivio di esempi di architettura esagonale GitHub Lambda.

Best practice

Per utilizzare questo pattern in un ambiente di produzione, segui queste best practice:

- Utilizza le chiavi gestite dai clienti in AWS Key Management Service (AWS KMS) per crittografare i gruppi di log di Amazon e le CloudWatch tabelle Amazon DynamoDB.
- Configura AWS WAF per Amazon API Gateway per consentire l'accesso solo dalla rete della tua organizzazione.
- Prendi in considerazione altre opzioni per l'autorizzazione dell'API Gateway se IAM non soddisfa le tue esigenze. Ad esempio, puoi utilizzare i pool di utenti di Amazon Cognito o gli autorizzatori API Gateway Lambda.
- Usa i backup DynamoDB.
- Configura le funzioni Lambda con un'implementazione di cloud privato virtuale (VPC) per mantenere il traffico di rete all'interno del cloud.
- Aggiorna la configurazione di origine consentita per il preflight CORS (Cross-Origin Resource Sharing) per limitare l'accesso solo al dominio di origine richiedente.

- Usa cdk-nag per controllare il codice CDK di AWS per le migliori pratiche di sicurezza.
- Prendi in considerazione l'utilizzo di strumenti di scansione del codice per trovare problemi di sicurezza comuni nel codice. Ad esempio, Bandit è uno strumento progettato per trovare problemi di sicurezza comuni nel codice Python. PIP-Audit analizza gli ambienti Python alla ricerca di pacchetti che presentano vulnerabilità note.

Questo modello utilizza AWS X-Ray per tracciare le richieste attraverso il punto di ingresso, il dominio e gli adattatori dell'applicazione. AWS X-Ray aiuta gli sviluppatori a identificare i colli di bottiglia e determinare latenze elevate per migliorare le prestazioni delle applicazioni.

Epiche

Inizializza il progetto

Operazione	Description	Competenze richieste
Crea il tuo repository.	1. Accedi a. GitHub 2. Crea un nuovo repository. Per istruzioni, consulta la GitHub documentazione. 3. Clona e trasferisci il repository di esempio per questo pattern nel nuovo repository del tuo account.	Sviluppatore di app
Installare le dipendenze.	1. Installa Poetry. `pip install poetry` 2. Installa i pacchetti dalla directory principale. Il comando seguente installa l'applicazione e i pacchetti AWS CDK. Installa anche i pacchetti di sviluppo necessari per l'esecuzi one dei test unitari. Tutti i	Sviluppatore di app

Operazione	Description	Competenze richieste

	pacchetti installati vengono collocati in un nuovo ambiente virtuale.
	`poetry install`
	3. Per vedere una rappresentazione grafica dei pacchetti installati, esegui il comando seguente.
	`poetry show --tree`
	4. Aggiorna tutte le dipendenze.
	`poetry update`
	5. Apri una nuova shell all'interno dell'ambiente virtuale appena creato. Contiene tutte le dipendenze installate.
	`poetry shell`	

Operazione	Description	Competenze richieste
Configura il tuo IDE.	Consigliamo Visual Studio Code, ma puoi usare qualsiasi IDE di tua scelta che supporti Python. I passaggi seguenti si riferiscono a Visual Studio Code. 1. Aggiorna il `.vscode/settings` file. `{ "python.testing.pytestArgs": [ "app/adapters/tests", "app/entrypoints/api/tests", "app/domain/tests" ], "python.testing.unittestEnabled": false, "python.testing.pytestEnabled": true, "python.envFile": "${workspaceFolder}/.env", }` 2. Crea un `.env` file nella directory principale del progetto. Questo assicura che la directory principale del progetto sia inclusa nella cartella principale, in `PYTHONPATH` modo che	Sviluppatore di app

Operazione	Description	Competenze richieste
	`pytest` possa trovarla e scoprire correttamente tutti i pacchetti. `PYTHONPATH=.`	
Esegui test unitari, opzione 1: usa Visual Studio Code.	1. Scegli l'interprete Python dell'ambiente virtuale gestito da Poetry. 2. Esegui i test da Test Explorer.	Sviluppatore di app

Esegui test unitari, opzione 2: usa i comandi della shell.	1. Avvia una nuova shell all'interno dell'ambiente virtuale.
	`poetry shell`
	2. Esegui il `pytest` comando dalla directory principale.
	`python -m pytest`
	In alternativa puoi eseguire il comando direttamente da Poetry.
	`poetry run python -m pytest`
		Sviluppatore di app

Implementate e testate l'applicazione

Operazione	Description	Competenze richieste
Richiedi credenziali temporanee.	Per avere credenziali AWS sulla shell durante l'esecuzione `cdk deploy`, crea credenziali temporanee utilizzando AWS IAM Identity Center (successore di AWS Single Sign-On). Per istruzioni, consulta il post sul blog Come recuperare le credenziali a breve termine per l'uso della CLI con AWS IAM Identity Center.	Sviluppatore di app, AWS DevOps

Distribuire l'applicazione.	1. Installa AWS CDK v2.
	`npm install -g aws-cdk`
	Per ulteriori informazioni, consulta la documentazione di AWS CDK.
	2. Esegui il bootstrap di AWS CDK nel tuo account e nella tua regione.
	`cdk bootstrap aws://12345678900/us-east-1 --profile aws-profile-name`
	3. Distribuisci l'applicazione come CloudFormation stack AWS utilizzando un profilo AWS.
		Sviluppatore di app, AWS DevOps

Operazione	Description	Competenze richieste
	`cdk deploy --profile aws-profile-name`	
Prova l'API, opzione 1: usa la console.	Utilizza la console API Gateway per testare l'API. Per ulteriori informazioni sulle operazioni e sui request/response messaggi dell'API, consulta la sezione sull'utilizzo dell'API del file readme nel GitHub repository.	Sviluppatore di app, AWS DevOps

Operazione	Description	Competenze richieste
Prova l'API, opzione 2: usa Postman.	Se vuoi usare uno strumento come Postman: 1. Installa Postman come applicazione autonoma o estensione del browser. 2. Copia l'URL dell'endpoint per l'API Gateway. Sarà nel seguente formato. `https://{api-id}.execute-api.{region}.amazonaws.com/{stage}/{path}` 3. Configura la firma AWS nella scheda di autorizzazione. Per istruzioni, consulta l'articolo AWS re:Post sull'attivazione dell'autenticazione IAM per API Gateway REST. APIs 4. Usa Postman per inviare richieste al tuo endpoint API.	Sviluppatore di app, AWS DevOps

Sviluppa il servizio

Operazione	Description	Competenze richieste
Scrivi test unitari per il dominio aziendale.	1. Crea un file Python nella `app/domain/tests` cartella usando il prefisso del nome del `test_` file.	Sviluppatore di app

Operazione	Description	Competenze richieste

	2. Crea un nuovo metodo di test per testare la nuova logica aziendale utilizzando l'esempio seguente.
	
	`def test_create_product_should_store_in_repository():`
	`    # Arrange`
	`    command = create_product_command.CreateProductCommand(`
	`        name="Test Product",`
	`        description="Test Description",`
	`    )`
	`    # Act`
	`    create_product_command_handler.handle_create_product_command(`
	`        command=command, unit_of_work=mock_unit_of_work`
	`    )`
	`# Assert`
	
	3. Create una classe di comando nella `app/domain/commands` cartella.
	4. Se la funzionalità è nuova, create uno stub per il gestore dei comandi nella	

Operazione	Description	Competenze richieste
	`app/domain/command_handlers` cartella. 5. Eseguite lo unit test per verificarne l'esito negativo, poiché non esiste ancora alcuna logica aziendale. `python -m pytest`	

Operazione	Description	Competenze richieste

Implementa comandi e gestori di comandi.	1. Implementa la logica aziendale nel file del gestore dei comandi appena creato.
	2. Per ogni dipendenza che interagisce con sistemi esterni, dichiarate una classe astratta nella cartella. `app/domain/ports`
	
	`class ProductsRepository(ABC):`
	`  @abstractmethod`
	`  def add(self, product: product.Product) -> None:`
	`  ...`
	
	`class UnitOfWork(ABC):`
	`  products: ProductsRepository`
	
	`  @abstractmethod`
	`  def commit(self) -> None:`
	`  ...`
	
	`  @abstractmethod`
	`  def __enter__(self) -> typing.Any:`
	`  ...`
	
	`  @abstractmethod`
	`  def __exit__(self, *args) -> None:`
		Sviluppatore di app

Operazione	Description	Competenze richieste

	<pre>...</pre>
	3. Aggiorna la firma del gestore dei comandi per accettare le nuove dipendenze dichiarate utilizzando la classe di porta astratta come annotazione del tipo.
	<pre>def handle_cr
	eate_product_comma
	nd(
	    command: create_pr
	oduct_command.Crea
	teProductCommand,
	    unit_of_work:
	 unit_of_work.UnitO
	fWork,
	) -> str:
	    ...</pre>
	4. Aggiorna lo unit test per simulare il comportam ento di tutte le dipendenze dichiarate per il gestore dei comandi.
	<pre> # Arrange
	    mock_unit_of_work
	 = unittest.mock.crea
	te_autospec(
	        spec=unit
	_of_work.UnitOfWor
	k, instance=True
	    )
	    mock_unit
	_of_work.products =</pre>	

Operazione	Description	Competenze richieste
| | ```
 unittest.mock.crea
te_autospec(
        spec=unit
_of_work.ProductsR
epository, instance=
True
    )
``` | |
| | 5. Aggiorna la logica di asserzione nel test per verificare le chiamate di dipendenza previste.<br>```
  # Assert
    mock_unit
_of_work.commit.as
sert_called_once()
    product =
 mock_unit_of_work.
products.add.call_
args.args[0]

    assertpy.
assert_that(produc
t.name).is_equal_t
o("Test Product")
    assertpy.
assert_that(produc
t.description).is_
equal_to("Test
 Description")
``` | |
| | 6. Esegui il test unitario per verificarne l'esito positivo.<br>```
python -m pytest
``` | |
```

Operazione	Description	Competenze richieste
Scrivi test di integrazione per adattatori secondari.	1. Crea un file di test nella `app/adapters/tests` cartella utilizzando `test_` come nome di file il prefisso. 2. Usa la libreria Moto per simulare i servizi AWS. 3. Crea un nuovo metodo di test per un test di integrazione dell'adattatore.	Sviluppatore di app

```
@pytest.fixture
def mock_dynamodb():
   with moto.mock
_dynamodb():
    yield boto3.res
ource("dynamodb",
 region_name="eu-ce
ntral-1")
```

```
def test_add_
and_commit_should_
store_product(mock
_dynamodb):
    # Arrange
    unit_of_work =
 dynamodb_unit_of_w
ork.DynamoDBUnitOf
Work(
        table_nam
e=TEST_TABLE_NAME,
 dynamodb_client=mo
ck_dynamodb.meta.c
lient
    )
    current_time =
 datetime.datetime.
now(datetime.timez
```

Operazione	Description	Competenze richieste

```
one.utc).isoformat
()

    new_product_id =
 str(uuid.uuid4())
    new_product =
 product.Product(
        id=new_pr
oduct_id,
        name="test-
name",
        descripti
on="test-descripti
on",
        createDat
e=current_time,
        lastUpdat
eDate=current_time,
    )

    # Act
    with unit_of_w
ork:
        unit_of_w
ork.products.add(n
ew_product)
        unit_of_w
ork.commit()

    # Assert
```

4. Crea una classe di adattatore nella `app/adapters` cartella. Utilizzate la classe astratta della cartella ports come classe base.
5. Esegui il test unitario per vederlo fallire, perché non c'è ancora alcuna logica.

Operazione	Description	Competenze richieste
	`python -m pytest`	

Operazione	Description	Competenze richieste

Implementa adattatori secondari.	1. Implementa la logica nel file dell'adattatore appena creato.
	2. Aggiorna le asserzioni di test.
	
	`# Assert`
	`    with unit_of_work_readonly:`
	`        product_from_db = unit_of_work_readonly.products.get(new_product_id)`
	
	`    assertpy.assert_that(product_from_db).is_not_none()`
	`    assertpy.assert_that(product_from_db.dict()).is_equal_to(`
	`        {`
	`            "id": new_product_id,`
	`            "name": "test-name",`
	`            "description": "test-description",`
	`            "createDate": current_time,`
	`            "lastUpdateDate": current_time,`
	`        }`
	`    )`
		Sviluppatore di app

Operazione	Description	Competenze richieste
	3. Esegui il test unitario per verificarne l'esito positivo. `python -m pytest`	

Operazione	Description	Competenze richieste
Scrivi end-to-end dei test.	1. Crea un file di test nella `app/entrypoints/api/tests` cartella utilizzando `test_` come prefisso del nome file. 2. Crea un dispositivo di contesto Lambda che verrà utilizzato dal test per chiamare Lambda. 3. Crea un metodo di test per la chiamata all'API.	Sviluppatore di app

```
@pytest.fixture
def lambda_co
ntext():
    @dataclass
    class LambdaCon
text:
        function_
name: str = "test"
        memory_li
mit_in_mb: int = 128
        invoked_f
unction_arn: str =
 "arn:aws:lambda:eu
-west-1:809313241:
function:test"
        aws_reque
st_id: str =
 "52fdfc07-2182-154
f-163f-5f0f9a621d7
2"

 return LambdaCon
text()
```

Operazione	Description	Competenze richieste

	`def test_crea te_product(lambda_ context):`
	`  # Arrange`
	`  name = "TestName"`
	`  description = "Test description"`
	`  request = api_model.CreatePr oductRequest(name= name, descripti on=description)`
	
	`  minimal_event = api_gateway_proxy_ event.APIGatewayPr oxyEvent(`
	`  {`
	`  "path": "/ products",`
	`  "httpMeth od": "POST",`
	`  "requestC ontext": {  # correlation ID`
	
	`  "requestId": "c6af9ac6-7b61-11e 6-9a41-93e8deadbee f"`
	`  },`
	`  "body": json.dumps(request .dict()),`
	`  }`
	`  )`
	
	`  create_pr oduct_func_mock = unittest.mock.crea te_autospec(`	

Operazione	Description	Competenze richieste

	<pre>spec=crea
	te_product_command
	_handler.handle_cr
	eate_product_comma
	nd
	)
	handler.c
	reate_product_comm
	and_handler.handle
	_create_product_co
	mmand = (
	create_pr
	oduct_func_mock
	)
	
	# Act
	handler.h
	andler(minimal_eve
	nt, lambda_context)</pre>	

	4. Esegui lo unit test per vederlo fallire, perché non c'è ancora alcuna logica. <pre>python -m pytest</pre>	

Operazione	Description	Competenze richieste

Implementa gli adattatori primari.	1. Crea una funzione per la logica di business dell'API e dichiarala come risorsa API.
	
	`@tracer.capture_method`
	`@app.post("/products")`
	`@utils.parse_event(model=api_model.CreateProductRequest, app_context=app)`
	`def create_product(`
	`request: api_model.CreateProductRequest,`
	`) -> api_model.CreateProductResponse:`
	`"""Creates a product."""`
	`...`
	
	Note
	Tutti i decoratori che vedi sono funzionalità della libreria AWS Lambda Powertools for Python. Per i dettagli, consulta il sito Web AWS Lambda Powertools for Python.
		Sviluppatore di app

Operazione	Description	Competenze richieste

	2. Implementa la logica dell'API.
	
	`id=create_product_command_handler.handle_create_product_command( command=create_product_command.CreateProductCommand( name=request.name, description=request.description, ), unit_of_work=unit_of_work, ) response = api_model.CreateProductResponse(id=id) return response.dict()`
	
	3. Esegui lo unit test per verificarne l'esito positivo.
	
	`python -m pytest`	

Risorse correlate

Guida APG

- Creazione di architetture esagonali su AWS

Riferimenti AWS

- Documentazione AWS Lambda
- Documentazione CDK AWS
  - La tua prima app AWS CDK
- Documentazione API Gateway
  - Controlla l'accesso a un'API con autorizzazioni IAM
  - Usa la console API Gateway per testare un metodo API REST
- Documentazione di Amazon DynamoDB

Strumenti

- Sito web git-scm.com
- Installazione di Git
- Creare un nuovo GitHub repository
- Sito web Python
- AWS Lambda Powertools per Python
- Sito web Postman
- Libreria di oggetti fittizi in Python
- Sito web di poesia

IDEs

- Sito Web di Visual Studio Code
- PyCharm website

Altri modelli

- Accedi alle applicazioni container in modo privato su Amazon ECS utilizzando AWS Fargate, PrivateLink AWS e un Network Load Balancer
- Accedi alle applicazioni container in modo privato su Amazon ECS utilizzando AWS PrivateLink e un Network Load Balancer
- Associa un AWS CodeCommit repository in uno Account AWS ad Amazon SageMaker AI Studio Classic in un altro account
- Eliminazione automatica degli AWS CloudFormation stack e delle risorse associate
- Limita l'accesso in base all'indirizzo IP o alla geolocalizzazione utilizzando AWS WAF
- Crea un'app mobile React Native senza server utilizzando AWS Amplify
- Crea e testa app iOS con AWS CodeCommit, AWS e CodePipeline AWS Device Farm
- Configura la registrazione per le applicazioni.NET in Amazon CloudWatch Logs utilizzando NLog
- Consolida la generazione di URL predefiniti di Amazon S3 e il download di oggetti utilizzando un endpoint associato a indirizzi IP statici
- Crea una definizione di attività Amazon ECS e monta un file system su EC2 istanze utilizzando Amazon EFS
- Implementa un'applicazione basata su gRPC su un cluster Amazon EKS e accedi ad essa con un Application Load Balancer
- Implementa una ChatOps soluzione per gestire i risultati della scansione SAST utilizzando Amazon Q Developer nelle applicazioni di chat, azioni personalizzate e CloudFormation
- Implementa i canarini CloudWatch Synthetics utilizzando Terraform
- Distribuisci microservizi Java su Amazon ECS utilizzando AWS Fargate
- Implementa uno use case RAG AWS utilizzando Terraform e Amazon Bedrock
- Distribuisci risorse in una AWS Wavelength zona utilizzando Terraform
- Implementa il controllo delle versioni delle API basato sul percorso utilizzando domini personalizzati in Amazon API Gateway
- Esegui la migrazione di una coda di messaggistica da Microsoft Azure Service Bus ad Amazon SQS
- Esegui la migrazione di un'applicazione.NET da Microsoft Azure App Service ad AWS Elastic Beanstalk
- Esegui la migrazione di un'applicazione web Go locale su AWS Elastic Beanstalk utilizzando il metodo binario

- Esegui la migrazione di un server SFTP locale a Using AWS AWS Transfer for SFTP
- Esegui la migrazione da IBM WebSphere Application Server ad Apache Tomcat su Amazon EC2
- Esegui la migrazione da IBM WebSphere Application Server ad Apache Tomcat su Amazon EC2 con Auto Scaling
- Esegui la migrazione di applicazioni Java locali su AWS utilizzando AWS App2Container
- Migrazione dei certificati SSL di Windows su un Application Load Balancer utilizzando ACM
- Modernizza le applicazioni ASP.NET Web Forms su AWS
- Monitora l'attività delle applicazioni utilizzando CloudWatch Logs Insights
- Esegui un contenitore Docker dell'API Web ASP.NET Core su un'istanza Amazon Linux EC2
- Invia attributi personalizzati ad Amazon Cognito e inseriscili in token
- Distribuisci contenuti statici in un bucket Amazon S3 tramite un VPC utilizzando Amazon CloudFront
- Configura un' PeopleSoft architettura ad alta disponibilità su AWS
- Semplifica lo sviluppo e la distribuzione di bot Amazon Lex utilizzando un flusso di lavoro automatizzato
- Risolvi i problemi relativi agli stati AWS Step Functions utilizzando Amazon Bedrock
- Usa Network Firewall per acquisire i nomi di dominio DNS dall'Indicazione del nome del server per il traffico in uscita
- Visualizza i risultati AI/ML del modello utilizzando Flask e AWS Elastic Beanstalk

IoT

Argomenti

- Estrarre e interrogare gli attributi AWS IoT SiteWise dei metadati in un data lake
- Configurazione e risoluzione dei problemi di AWS IoT Greengrass con dispositivi client
- Altri modelli

Estrarre e interrogare gli attributi AWS IoT SiteWise dei metadati in un data lake

Ambarish Dongaonkar, Amazon Web Services

Riepilogo

AWS IoT SiteWise utilizza modelli e gerarchie di asset per rappresentare le apparecchiature, i processi e le strutture industriali. Ogni modello o asset può avere più attributi specifici dell'ambiente. Gli attributi dei metadati di esempio includono il sito o l'ubicazione fisica dell'asset, i dettagli dell'impianto e gli identificatori delle apparecchiature. Questi valori di attributo integrano i dati di misurazione degli asset per massimizzare il valore aziendale. L'apprendimento automatico (ML) può fornire ulteriori informazioni su questi metadati e semplificare le attività di progettazione.

Tuttavia, gli attributi dei metadati non possono essere richiesti direttamente dal servizio. AWS IoT SiteWise Per rendere gli attributi interrogabili, è necessario estrarli e inserirli in un data lake. Questo modello utilizza uno script Python per estrarre gli attributi di tutte le AWS IoT SiteWise risorse e inserirli in un data lake in un bucket Amazon Simple Storage Service (Amazon S3). Una volta completato questo processo, puoi utilizzare le query SQL in Amazon Athena per accedere AWS IoT SiteWise agli attributi dei metadati e ad altri set di dati, come i set di dati di misurazione. Le informazioni sugli attributi dei metadati sono utili anche quando si lavora con monitor o dashboard. AWS IoT SiteWise Puoi anche creare un dashboard Amazon Quick Sight utilizzando gli attributi estratti nel bucket Amazon S3.

Il pattern ha un codice di riferimento e puoi implementarlo utilizzando i migliori servizi di elaborazione per il tuo caso d'uso, come o. AWS Lambda AWS Glue

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS.
- Autorizzazioni per configurare AWS Lambda funzioni o AWS Glue lavori.
- Un bucket Amazon S3.
- I modelli e le gerarchie degli asset sono configurati in. AWS IoT SiteWise Per ulteriori informazioni, consulta Creazione di modelli di asset nella AWS IoT SiteWise documentazione.

Architecture

È possibile utilizzare una funzione Lambda o un AWS Glue job per completare questo processo. Ti consigliamo di utilizzare Lambda se hai meno di 100 modelli e ogni modello ha una media di 15 o meno attributi. Per tutti gli altri casi d'uso, consigliamo di utilizzare AWS Glue.

L'architettura della soluzione e il flusso di lavoro sono illustrati nel diagramma seguente.

1. Viene eseguito il AWS Glue job pianificato o la funzione Lambda. Estrae gli attributi dei metadati degli asset AWS IoT SiteWise e li inserisce in un bucket Amazon S3.
2. Un AWS Glue crawler esegue la scansione dei dati estratti nel bucket Amazon S3 e crea tabelle in un. AWS Glue Data Catalog
3. Utilizzando SQL standard, Amazon Athena esegue query sulle tabelle in. AWS Glue Data Catalog

Automazione e scalabilità

È possibile pianificare l'esecuzione giornaliera o settimanale della funzione o del AWS Glue processo Lambda, in base alla frequenza di aggiornamento delle configurazioni AWS IoT SiteWise degli asset.

Non c'è limite al numero di AWS IoT SiteWise risorse che il codice di esempio può elaborare, ma un numero elevato di risorse può aumentare il tempo necessario per completare il processo.

Tools (Strumenti)

- Amazon Athena è un servizio di query interattivo che ti aiuta ad analizzare i dati direttamente in Amazon S3 utilizzando SQL standard.
- AWS Glueè un servizio di estrazione, trasformazione e caricamento (ETL) completamente gestito. Ti aiuta a classificare, pulire, arricchire e spostare i dati in modo affidabile tra archivi di dati e flussi di dati.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS IoT SiteWiseti aiuta a raccogliere, modellare, analizzare e visualizzare i dati delle apparecchiature industriali su larga scala.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.

- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS SDK per Python (Boto3)è un kit di sviluppo software che ti aiuta a integrare la tua applicazione, libreria o script Python con. Servizi AWS

Epiche

Imposta il lavoro o la funzione

Operazione	Description	Competenze richieste
Configura le autorizzazioni in IAM.	Nella console IAM, concedi le autorizzazioni al ruolo IAM assunto dalla funzione AWS Glue o dal job Lambda per eseguire le seguenti operazioni: • Leggi dal servizio AWS IoT SiteWise • Scrivi nel bucket Amazon S3 Per ulteriori informazioni, consulta Creating a role for an Servizio AWS nella documentazione IAM.	Informazioni generali su AWS
Crea la funzione o AWS Glue il job Lambda.	Se utilizzi Lambda, crea una nuova funzione Lambda. Per Runtime, scegli Python. Per ulteriori informazioni, consulta Creazione di funzioni Lambda con Python nella documentazione di Lambda.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	Se lo stai usando AWS Glue, crea un nuovo job della shell Python nella AWS Glue console. Per ulteriori informazioni, consulta Aggiungere lavori di shell Python nella AWS Glue documentazione.	
Aggiorna la funzione o AWS Glue il job Lambda.	Modifica la nuova funzione o AWS Glue job Lambda e inserisci l'esempio di codice nella sezione Informazioni aggiuntive. Modifica il codice in base alle esigenze del tuo caso d'uso. Per ulteriori informazioni, consulta Modificare il codice utilizzando l'editor della console nella documentazione di Lambda e vedere Lavorare con gli script nella documentazione.AWS Glue	Informazioni generali su AWS

Esegui il lavoro o la funzione

Operazione	Description	Competenze richieste
Esegui la funzione o AWS Glue il processo Lambda.	Esegui la funzione o AWS Glue il processo Lambda. Per ulteriori informazioni, consulta Invocare la funzione Lambda nella documentazione di Lambda o vedere Avvio di lavori utilizzando i trigger nella documentazione.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	AWS Glue Questo estrae gli attributi dei metadati per gli asset e i modelli nella AWS IoT SiteWise gerarchia e li archivia nel bucket Amazon S3 specificato.	
Configura un AWS Glue crawler.	Configura un AWS Glue crawler con il classificatore di formato necessario per un file in formato CSV. Usa il bucket Amazon S3 e i dettagli del prefisso utilizzat i nella funzione o nel job Lambda. AWS Glue Per ulteriori informazioni, consulta Definizione dei crawler nella documentazione. AWS Glue	Informazioni generali su AWS
Esegui il AWS Glue crawler.	Esegui il crawler per elaborare il file di dati creato dalla funzione o dal processo Lambda. AWS Glue Il crawler crea una tabella nel campo specificato. AWS Glue Data Catalog Per ulteriori informazi oni, consulta Avvio dei crawler utilizzando i trigger nella documentazione. AWS Glue	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Interroga gli attributi dei metadati.	Utilizzando Amazon Athena, usa SQL standard per eseguire query in base alle esigenze del AWS Glue Data Catalog tuo caso d'uso. Puoi unire la tabella degli attributi dei metadati ad altri database e tabelle. Per ulteriori informazioni, consulta la sezione Guida introduttiva nella documentazione di Amazon Athena.	Informazioni generali su AWS

Risorse correlate

- Documentazione Amazon Athena
- AWS Glue documentazione
- AWS IoT SiteWise Documentazione di riferimento dell'API
- AWS IoT SiteWise guida per l'utente
  - Nozioni di base
  - Modellazione di asset industriali
  - Definizione delle relazioni tra i modelli di asset (gerarchie)
  - Associazione e dissociazione degli asset
  - Creazione della demo AWS IoT SiteWise
- IOTSiteWise (documentazione SDK per Python)
- Documentazione Lambda

Informazioni aggiuntive

Codice

Il codice di esempio fornito è di riferimento ed è possibile personalizzare questo codice in base alle esigenze del caso d'uso.

```
# Following code can be used in an AWS Lambda function or in an AWS Glue Python shell
 job.
# IAM roles used for this job need read access to the AWS IoT SiteWise service and
 write access to the S3 bucket.
sw_client = boto3.client('iotsitewise')
s3_client = boto3.client('s3')
output = io.StringIO()

attribute_list=[]
bucket = '{3_bucket name}'
prefix = '{s3_bucket prefix}'
output.write("model_id,model_name,asset_id,asset_name,attribuet_id,attribute_name,attribute_val
\n")

m_resp = sw_client.list_asset_models()
for m_rec in m_resp['assetModelSummaries']:
     model_id = m_rec['id']
     model_name = m_rec['name']

     attribute_list.clear()
     dam_response = sw_client.describe_asset_model(assetModelId=model_id)
     for rec in dam_response['assetModelProperties']:
         if 'attribute' in rec['type']:
            attribute_list.append(rec['name'])

     response = sw_client.list_assets(assetModelId=model_id, filter='ALL')
     for asset in response['assetSummaries']:
         asset_id = asset['id']
         asset_name = asset['name']
         resp = sw_client.describe_asset(assetId=asset_id)
         for rec in resp['assetProperties']:
            if rec['name'] in attribute_list:
                p_resp = sw_client.get_asset_property_value(assetId=asset_id,
 propertyId=rec['id'])
                if 'propertyValue' in p_resp:
                    if p_resp['propertyValue']['value']:
                        if 'stringValue' in p_resp['propertyValue']['value']:
                             output.write(model_id + "," + model_name + ","
 + asset_id + "," + asset_name + "," + rec['id'] + "," + rec['name'] + "," +
```

```
 str(p_resp['propertyValue']['value']['stringValue']) + "\n")

                        if 'doubleValue' in p_resp['propertyValue']['value']:
                             output.write(model_id + "," + model_name + ","
 + asset_id + "," + asset_name + "," + rec['id'] + "," + rec['name'] + "," +
 str(p_resp['propertyValue']['value']['doubleValue']) + "\n")
                        if 'integerValue' in p_resp['propertyValue']['value']:
                             output.write(model_id + "," + model_name + ","
 + asset_id + "," + asset_name + "," + rec['id'] + "," + rec['name'] + "," +
 str(p_resp['propertyValue']['value']['integerValue']) + "\n")
                         if 'booleanValue' in p_resp['propertyValue']['value']:
                             output.write(model_id + "," + model_name + ","
 + asset_id + "," + asset_name + "," + rec['id'] + "," + rec['name'] + "," +
 str(p_resp['propertyValue']['value']['booleanValue']) + "\n")

output.seek(0)
s3_client.put_object(Bucket=bucket, Key= prefix + '/data.csv', Body=output.getvalue())
output.close()
```

Configurazione e risoluzione dei problemi di AWS IoT Greengrass con dispositivi client

Marouane Sefiani e Akalanka De Silva, Amazon Web Services

Riepilogo

AWS IoT Greengrass è un servizio cloud e di runtime edge open source per la creazione, la distribuzione e la gestione di software Internet of Things (IoT) su dispositivi edge. I casi d'uso per AWS IoT Greengrass includono:

- Case intelligenti in cui un gateway AWS IoT Greengrass viene utilizzato come hub per l'automazione domestica
- Fabbriche intelligenti in cui AWS IoT Greengrass può facilitare l'acquisizione e l'elaborazione locale dei dati dall'officina

AWS IoT Greengrass può fungere da endpoint di connessione MQTT sicuro e autenticato per altri dispositivi edge (noti anche come dispositivi client), che altrimenti si connetterebbero direttamente ad AWS IoT Core. Questa funzionalità è utile quando i dispositivi client non hanno accesso diretto alla rete all'endpoint AWS IoT Core.

Puoi configurare AWS IoT Greengrass per l'uso con i dispositivi client per i seguenti casi d'uso:

- Per consentire ai dispositivi client di inviare dati ad AWS IoT Greengrass
- Per l'inoltro dei dati ad AWS IoT Core da parte di AWS IoT Greengrass
- Per sfruttare le funzionalità avanzate del motore di regole AWS IoT Core

Queste funzionalità richiedono l'installazione e la configurazione dei seguenti componenti sul dispositivo AWS IoT Greengrass:

- Broker MQTT
- Ponte MQTT
- Autenticazione del dispositivo client
- Rilevatore IP

Inoltre, i messaggi pubblicati dai dispositivi client devono essere in formato JSON o in formato Protocol Buffers (protobuf).

Questo modello descrive come installare e configurare questi componenti richiesti e fornisce suggerimenti e best practice per la risoluzione dei problemi.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- AWS Command Line Interface (AWS CLI) versione 2
- Due dispositivi client che eseguono Python 3.7 o versione successiva
- Un dispositivo principale che esegue Java Runtime Environment (JRE) versione 8 o successiva e Amazon Corretto 11 o OpenJDK 11

Limitazioni

- Devi scegliere una regione AWS in cui è disponibile AWS IoT Core. Per l'elenco aggiornato delle regioni per AWS IoT Core, consulta Servizi AWS per regione.
- Il dispositivo principale deve avere almeno 172 MB di RAM e 512 MB di spazio su disco.

Architecture

Il diagramma seguente mostra l'architettura della soluzione per questo modello.

L'architettura include:

- Due dispositivi client. Ogni dispositivo contiene una chiave privata, un certificato del dispositivo e un certificato di autorità di certificazione (CA) principale. L'SDK per dispositivi AWS IoT, che contiene un client MQTT, è installato anche su ogni dispositivo client.
- Un dispositivo principale su cui è distribuito AWS IoT Greengrass con i seguenti componenti:
  - Broker MQTT
  - Ponte MQTT
  - Autenticazione del dispositivo client
  - Rilevatore IP

Questa architettura supporta i seguenti scenari:

- I dispositivi client possono utilizzare il proprio client MQTT per comunicare tra loro tramite il broker MQTT del dispositivo principale.
- I dispositivi client possono anche comunicare con AWS IoT Core nel cloud tramite il broker MQTT del dispositivo principale e il bridge MQTT.
- AWS IoT Core nel cloud può inviare messaggi ai dispositivi client tramite il client di test MQTT e il bridge MQTT e il broker MQTT del dispositivo principale.

Per ulteriori informazioni sulle comunicazioni tra i dispositivi client e il dispositivo principale, consulta la sezione Informazioni aggiuntive.

Tools (Strumenti)

Servizi AWS

- AWS IoT Greengrass è un servizio cloud e di runtime edge open source per l'Internet of Things (IoT) che ti aiuta a creare, distribuire e gestire applicazioni IoT sui tuoi dispositivi.
- AWS IoT Core fornisce comunicazioni sicure e bidirezionali per i dispositivi connessi a Internet per connettersi al cloud AWS.
- AWS IoT Device SDK è un kit di sviluppo software che include librerie open source, guide per sviluppatori con esempi e guide al porting per creare prodotti o soluzioni IoT innovativi su piattaforme hardware a tua scelta.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.

Best practice

- Il payload dei messaggi dai dispositivi client deve essere in formato JSON o Protobuf per sfruttare le funzionalità avanzate del motore di regole AWS IoT Core, come la trasformazione e le azioni condizionali.
- Configura il bridge MQTT per consentire la comunicazione bidirezionale.
- Configura e distribuisci il componente del rilevatore IP in AWS IoT Greengrass per garantire che gli indirizzi IP del dispositivo principale siano inclusi nel campo Subject Alternative Name (SAN) del certificato broker MQTT.

Epiche

Configura il dispositivo principale

Operazione	Description	Competenze richieste
Configura AWS IoT Greengrass sul tuo dispositivo principale.	Installa il software AWS IoT Greengrass Core seguendo le istruzioni nella guida per sviluppatori.	AWS IoT Greengrass

Controlla lo stato dell'installazione.	Utilizza il seguente comando per verificare lo stato del servizio AWS IoT Greengrass sul tuo dispositivo principale:
	`sudo systemctl status greengrass.service`
	L'output previsto del comando è:
	`Launched Nucleus successfully`
		Informazioni generali su AWS

Imposta una policy IAM e collegala al ruolo di servizio Greengrass.	1. Crea una policy IAM per consentire le comunicazioni da e verso il bridge MQTT. Ecco un esempio di policy: `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow",`	Informazioni generali su AWS

Operazione	Description	Competenze richieste

	<pre>"Action":
	[
	
	"iot:*"
	],
	"Resource
	": "*"
	},
	{
	"Sid":
	"GreengrassActions
	",
	"Effect":
	"Allow",
	"Action":
	[
	
	"greengrass:*"
	],
	"Resource
	": "*"
	}
	]
	}</pre>	

	2. Associa la policy al ruolo di servizio Greengrass. Per ottenere il ruolo di servizio, usa il comando:
	
	<pre>aws greengrassv2
	get-service-role-f
	or-account --region
	<region></pre>
	dove <region> si riferisce alla tua regione AWS.	

Operazione	Description	Competenze richieste
Configura e distribuisci i componenti richiesti nel dispositivo principale di AWS IoT Greengrass.	Configura e distribuisci i seguenti componenti: • `greengrass.clientdevices.mqtt.Moquette` (vedi i dettagli di configurazione) • `greengrass.clientdevices.mqtt.Bridge` (vedi i dettagli di configurazione e l'attività successiva) • `greengrass.clientdevices.Auth` (vedi i dettagli di configurazione e l'attività successiva a quella successiva) • `aws.greengrass.clientdevices.IPDetector` (vedi i dettagli di configurazione)	AWS IoT Greengrass

Operazione	Description	Competenze richieste
Verificate che il bridge MQTT consenta la comunicazione bidirezionale.	Per inoltrare messaggi MQTT tra dispositivi client e AWS IoT Core, configura e distribuisci il componente bridge MQTT e specifica gli argomenti da inoltrare. Ecco un esempio: `{ "mqttTopicMapping": { "ClientDevicesToCloud": { "topic": "dt/#", "source": "LocalMqtt", "target": "IotCore" }, "CloudToClientDevices": { "topic": "cmd/#", "source": "IotCore", "target": "LocalMqtt" } } }`	AWS IoT Greengrass

Operazione	Description	Competenze richieste
Verifica che il componente di autenticazione consenta ai dispositivi client di connettersi e pubblicare o sottoscrivere argomenti.	La seguente `aws.greengrass.clientdevices.Auth` configurazione consente a tutti i dispositivi client di connettersi, pubblicare messaggi e sottoscrivere tutti gli argomenti. `{ "deviceGroups": { "formatVersion": "2021-03-05", "definitions": { "MyPermissiveDeviceGroup": { "selectionRule": "thingName: *", "policyName": "MyPermissivePolicy" } }, "policies": { "MyPermissivePolicy": { "AllowAll": { "statementDescription": "Allow client devices to perform all actions.", "operations": ["*"], "resources": ["*"] } }`	AWS IoT Greengrass

Operazione	Description	Competenze richieste

	<pre>}
	}
	}</pre>	

Configurare i dispositivi client

Operazione	Description	Competenze richieste
Installa l'SDK per dispositivi AWS IoT.	Installa l'SDK per dispositivi AWS IoT sui dispositivi client. Per un elenco completo delle lingue supportate e associate SDKs, consulta la documentazione di AWS IoT Core. Ad esempio, l'SDK per dispositivi AWS IoT per Python si trova su. GitHub Per installare questo SDK: 1. Verifica che Python 3.7 o versione successiva sia installato, come indicato nella pagina Prerequisiti del repository. GitHub 2. Usa il comando pip per installare l'SDK. Per macOS e Linux: <pre>python3 -m pip install awsiotsdk</pre> Per Windows:	Informazioni generali su AWS IoT

Operazione	Description	Competenze richieste

	`python -m pip install awsiotsdk`
	
	In alternativa, puoi installare l'SDK dal repository di origine:
	
	`# Create a workspace directory to hold all the SDK files`
	`mkdir sdk-workspace`
	`cd sdk-workspace`
	`# Clone the repository`
	`git clone https://github.com/aws/aws-iot-device-sdk-python-v2.git`
	`# Install using Pip (use 'python' instead of 'python3' on Windows)`
	`python3 -m pip install ./aws-iot-device-sdk-python-v2`	

Operazione	Description	Competenze richieste
Crea una cosa.	1. Nella console AWS IoT, se viene visualizzato un pulsante Get started, selezionalo. Altrimenti, nel pannello di navigazione, scegli Sicurezza, Politiche. 2. Se viene visualizzata la finestra di dialogo Non hai ancora alcuna politica, scegli Crea una politica. In caso contrario, scegliere Create (Crea). 3. Inserisci un nome per la policy di AWS IoT (ad esempio,`ClientDevicePolicy`). 4. Nella sezione Aggiungi dichiarazioni, sostituisci la policy esistente con il seguente codice JSON. Sostituisci `<region>` e `<account>` inserisci la tua regione AWS e il numero di account AWS. `{ "Version": "2012-10-17", "Statement": [{ "Effect": "Allow", "Action": "iot:Connect", "Resource": "arn:aws:iot:region:account:client/*"`	AWS IoT Core

Operazione	Description	Competenze richieste
	`},{"Effect": "Allow","Action": "iot:Publish","Resource": "*"},{"Effect": "Allow","Action": "iot:Receive","Resource": "*"},{"Effect": "Allow","Action": "iot:Subscribe","Resource": "*"},{"Effect": "Allow","Action": ["iot:GetThingShadow","iot:UpdateThingShadow","iot:DeleteThingShadow"],"Resource": "arn:aws:iot:region:account:thing/*"}]`	

Operazione	Description	Competenze richieste
	<pre>}</pre> 5. Scegli Create (Crea). 6. Nella console AWS IoT, nel pannello di navigazione, scegli Manage, Things. 7. Se viene visualizzata la finestra di dialogo Non hai ancora nulla, scegli Registra un oggetto. In caso contrario, scegliere Create (Crea). 8. Nella pagina Creating AWS IoT things (Creazione di oggetti AWS IoT), scegli Create a single thing (Crea singolo oggetto). 9. Nella pagina Add your device to the device registry (Aggiungi il tuo dispositivo al registro dei dispositivi), immettere un nome per l'oggetto IoT, ad esempio `ClientDevice1` , quindi scegliere Next (Avanti). **Note** Non puoi modificare il nome di un oggetto dopo averlo creato. Per cambiare il nome, è necessario creare un nuovo elemento,	

Operazione	Description	Competenze richieste
	assegnargli il nuovo nome e quindi eliminare quello vecchio. 10. Nella pagina Add a certificate for your thing (Aggiungi un certificato per l'oggetto), scegli Create certificate (Crea certificato). 11. Scegliere i collegamenti Download (Scarica) per scaricare il certificato, la chiave privata e il certificato CA root. **Important** Questa è l'unica occasione per scaricare il certificato e la chiave privata. 12. Scegli Activate (Attiva) per attivare il certificato. Il certificato deve essere attivo affinché un dispositivo possa connettersi ad AWS IoT. 13. Scegliere Attach a policy (Collega policy). 14. Per Aggiungi una policy per il tuo oggetto ClientDev	

Operazione	Description	Competenze richieste
	icePolicy, scegli Register Thing.	
Scarica il certificato CA dal dispositivo principale Greengrass.	Se si prevede che il dispositivo core Greengrass funzioni in ambienti offline, è necessario rendere disponibile il certificato CA di base Greengrass al dispositivo client in modo che possa verificare il certificato del broker MQTT (rilasciato dalla CA principale di Greengrass). Pertanto, è importante ottenere una copia di questo certificato. Utilizza uno dei seguenti approcci per scaricare il certificato CA: • Se hai accesso di rete al dispositivo AWS IoT Greengrass dal tuo PC, accedi `https://<device IP>:8883` al tuo browser Web e visualizza il certificato del broker MQTT e il certificato CA. Puoi anche salvare il certificato CA sul dispositivo client. • In alternativa, puoi usare la riga di comando OpenSSL: `openssl s_client -showcerts -connect <device IP>:8883`	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Copia le credenziali nei dispositivi client.	Copia il certificato CA di base Greengrass, il certificato del dispositivo e la chiave privata nei dispositivi client.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Associa i dispositivi client al dispositivo principale.	Associa i dispositivi client a un dispositivo principale in modo che possano scoprire il dispositivo principale. I dispositivi client possono quindi utilizzare l'API di scoperta Greengrass per recuperare le informazioni di connettività e i certificati per i dispositivi principali associati . Per ulteriori informazioni, consulta Associare i dispositivi client nella documentazione di AWS IoT Greengrass. 1. Nella console AWS IoT Greengrass, scegli i dispositivi Core. 2. Scegli il dispositivo principal e da gestire. 3. Nella pagina dei dettagli del dispositivo principale, scegli la scheda Dispositivi client. 4. Nella sezione Dispositi vi client associati, scegli Associa dispositivi client. 5. Nella modalità Associa i dispositivi client al dispositi vo principale, procedi come segue per ogni dispositivo client da associare: a. Inserisci il nome dell'oggetto AWS IoT da	AWS IoT Greengrass

Operazione	Description	Competenze richieste
	associare come dispositivo client. b. Scegliere Aggiungi. 6. Selezionare Associate (Associa). I dispositivi client che hai associato possono ora utilizzare l'API di scoperta Greengrass per scoprire questo dispositivo principale.	

Inviare e ricevere dati

Operazione	Description	Competenze richieste
Invia dati da un dispositivo client a un altro dispositivo client.	Utilizzate il client MQTT del vostro dispositivo per pubblicare un messaggio sull'`dt/client1/sensor`argomento.	Informazioni generali su AWS
Invia dati dal dispositivo client ad AWS IoT Core.	Usa il client MQTT sul tuo dispositivo per pubblicare un messaggio sull'`dt/client1/sensor` argomento. Nel client di test MQTT, sottoscrivete l'argomento su cui il dispositivo invia i messaggi oppure abbonatevi a # per tutti gli argomenti (consultate i dettagli).	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Invia messaggi da AWS IoT Core ai dispositivi client.	Nella pagina del client di test MQTT, nella scheda Pubblica su un argomento, nel campo Nome argomento , inserisci il nome dell'argo mento del messaggio. In questo esempio, utilizzare `cmd/client1` per l'argomen to.	Informazioni generali su AWS

risoluzione dei problemi

Problema	Soluzione
Impossibile verificare l'errore del certificato del server	Questo errore si verifica quando il client MQTT non è in grado di verificare il certificato presentato dal broker MQTT durante l'handsha ke TLS. Il motivo più comune è che il client MQTT non dispone del certificato CA. Segui questi passaggi per assicurarti che il certificato CA venga fornito al client MQTT. 1. Se hai accesso di rete al dispositivo AWS IoT Greengrass dal tuo PC, accedi `https://<device IP>:8883` in una finestra del browser per visualizzare il certificato del broker MQTT e il certificato CA. Puoi anche salvare il certificato CA sul dispositivo client. In alternativa, usa la riga di comando OpenSSL:

Problema	Soluzione
	``` openssl s_client -showcerts -connect <device IP>:8883 ``` 2. Salva il contenuto dei certificati Moquette CA e Greengrass Core CA in file, quindi visualizza il contenuto decodificato utilizzando il comando: ``` openssl x509 -in <Name of CA>.pem -text ``` Il certificato Moquette CA dovrebbe mostrare il campo SAN come in questo esempio: ``` X509v3 Subject Alternative Name: IP Address:XXX.XXX.XXX.XXX, IP Address:127.0.0.1, DNS:localhost ```
Impossibile verificare l'errore relativo al nome del server	Questo errore si verifica quando il client MQTT non è in grado di verificare che si stia connettendo al server corretto. Il motivo più comune è che l'indirizzo IP del dispositivo Greengrass non è elencato nel campo SAN del certificato. Segui le istruzioni della soluzione precedente per ottenere il certificato del broker MQTT e verifica che il campo SAN contenga l'indirizzo IP del dispositivo AWS IoT Greengrass, come spiegato nella sezione Informazioni aggiuntive. In caso contrario, verifica che il componente del rilevatore IP sia installato correttamente e riavvia il dispositivo principale.

Problema	Soluzione
Impossibile verificare il nome del server solo durante la connessione da un dispositivo client integrato	Mbed TLS, che è una popolare libreria TLS utilizzata nei dispositivi integrati, attualmente supporta la verifica dei nomi DNS solo nel campo SAN del certificato, come mostrato nel codice della libreria Mbed TLS. Poiché il dispositivo principale non ha un nome di dominio proprio e dipende dall'indirizzo IP, i client TLS che utilizzano Mbed TLS non riusciranno a verificare il nome del server durante l'handshake TLS, causando un errore di connessione. Ti consigliamo di aggiungere la verifica dell'indirizzo IP SAN alla tua libreria TLS Mbed con la funzione x509_crt_check_san.

Risorse correlate

- Documentazione di AWS IoT Greengrass
- Documentazione di AWS IoT Core
- Componente del broker MQTT
- Componente bridge MQTT
- Componente di autenticazione del dispositivo client
- Componente del rilevatore IP
- SDK per dispositivi AWS IoT
- Implementazione di dispositivi client locali con AWS IoT Greengrass (post sul blog AWS)
- RFC 5280 — Certificato dell'infrastruttura a chiave pubblica Internet X.509 e profilo dell'elenco di revoca dei certificati (CRL)

Informazioni aggiuntive

Questa sezione fornisce informazioni aggiuntive sulle comunicazioni tra i dispositivi client e il dispositivo principale.

Il broker MQTT ascolta sulla porta 8883 del dispositivo principale un tentativo di connessione al client TLS. L'illustrazione seguente mostra un esempio di certificato server del broker MQTT.

Il certificato di esempio mostra i seguenti dettagli:

- Il certificato è rilasciato dalla CA AWS IoT Greengrass Core, che è locale e specifica per il dispositivo principale, ovvero funge da CA locale.
- Questo certificato viene ruotato automaticamente ogni settimana dal componente di autenticazione del client, come mostrato nella figura seguente. È possibile impostare questo intervallo nella configurazione del componente client auth.

- Il nome alternativo del soggetto (SAN) svolge un ruolo fondamentale nella verifica del nome del server sul lato client TLS. Aiuta il client TLS a garantire la connessione al server corretto e aiuta a evitare man-in-the-middle attacchi durante la configurazione della sessione TLS. Nel certificato di esempio, il campo SAN indica che questo server è in ascolto su localhost (il socket del dominio Unix locale) e l'interfaccia di rete ha l'indirizzo IP 192.168.1.12.

Il client TLS utilizza il campo SAN nel certificato per verificare che si stia connettendo a un server legittimo durante la verifica del server. Al contrario, durante un tipico handshake TLS tra un server HTTP e un browser, il nome di dominio nel campo Common Name (CN) o nel campo SAN viene utilizzato per verificare il dominio a cui il browser si sta effettivamente connettendo durante il processo di verifica del server. Se il dispositivo principale non ha un nome di dominio, l'indirizzo IP incluso nel campo SAN ha lo stesso scopo. Per ulteriori informazioni, vedere la sezione Subject Alternative Name di RFC 5280 — Profilo del certificato e dell'elenco di revoca dei certificati (CRL) dell'infrastruttura a chiave pubblica Internet X.509.

Il componente del rilevatore IP in AWS IoT Greengrass assicura che gli indirizzi IP corretti siano inclusi nel campo SAN del certificato.

Il certificato nell'esempio è firmato dal dispositivo AWS IoT Greengrass che funge da CA locale. Il client TLS (client MQTT) non è a conoscenza di questa CA, quindi dobbiamo fornire un certificato CA simile al seguente.

Altri modelli

- Inserimento conveniente di dati IoT direttamente in Amazon S3 con AWS IoT Greengrass
- Implementa sistemi agentici su Amazon Bedrock con il framework CrewAI utilizzando Terraform
- Distribuisci applicazioni containerizzate in AWS IoT Greengrass V2 esecuzione come contenitore Docker

Migrazione e modernizzazione

Argomenti

- Migrazione
- Modernizzazione
- Mainframe

Migrazione

Argomenti

- Crea CloudFormation modelli AWS per attività AWS DMS utilizzando Microsoft Excel e Python
- Inizia con l'individuazione automatica del portafoglio
- Esegui la migrazione dei carichi di lavoro Cloudera locali a Cloudera Data Platform su AWS
- Risolvi gli errori di connessione dopo la migrazione di Microsoft SQL Server al cloud AWS
- Riavvia automaticamente l'AWS Replication Agent senza disabilitarlo SELinux dopo il riavvio di un server di origine RHEL
- Re-architetto
- Riospitare
- Trasferisci
- Conversione piattaforma
- Modelli di migrazione per carico di lavoro
- Altri modelli

Crea CloudFormation modelli AWS per attività AWS DMS utilizzando Microsoft Excel e Python

Venkata Naveen Koppula, Amazon Web Services

Riepilogo

Questo modello descrive i passaggi per la creazione automatica di CloudFormation modelli AWS per AWS Database Migration Service (AWS DMS) utilizzando Microsoft Excel e Python.

La migrazione dei database con AWS DMS spesso implica la creazione di CloudFormation modelli AWS per il provisioning delle attività AWS DMS. In precedenza, la creazione di CloudFormation modelli AWS richiedeva la conoscenza del linguaggio di programmazione JSON o YAML. Con questo strumento, è necessaria solo una conoscenza di base di Excel e di come eseguire uno script Python utilizzando un terminale o una finestra di comando.

Come input, lo strumento utilizza una cartella di lavoro di Excel che include i nomi delle tabelle da migrare, Amazon Resource Names (ARNs) degli endpoint AWS DMS e le istanze di replica AWS DMS. Lo strumento genera quindi CloudFormation modelli AWS per le attività AWS DMS richieste.

Per passaggi dettagliati e informazioni di base, consulta il post del blog Create AWS CloudFormation templates for AWS DMS tasks using Microsoft Excel nel blog AWS Database.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Microsoft Excel versione 2016 o successiva
- Python versione 2.7 o successiva
- Il modulo Python xlrd (installato al prompt dei comandi con il comando: pip install xlrd)
- Endpoint di origine e destinazione di AWS DMS e istanza di replica AWS DMS

Limitazioni

- I nomi degli schemi, delle tabelle e delle colonne associate vengono trasformati in caratteri minuscoli negli endpoint di destinazione.
- Questo strumento non si occupa della creazione di endpoint e istanze di replica AWS DMS.

- Attualmente, lo strumento supporta solo uno schema per ogni attività AWS DMS.

Architecture

Stack tecnologico di origine

- Un database locale
- Microsoft Excel

Stack tecnologico Target

- CloudFormation Modelli AWS
- Un database nel cloud AWS

Architettura

Tools (Strumenti)

- Pycharm IDE o qualsiasi ambiente di sviluppo integrato (IDE) che supporti Python versione 3.6
- Microsoft Office 2016 (per Microsoft Excel)

Epiche

Configura la rete, l'istanza di replica AWS DMS e gli endpoint

Operazione	Description	Competenze richieste
Se necessario, richiedi un aumento della quota di servizio.	Richiedi un aumento della quota di servizio per le attività di AWS DMS, se necessario.	Informazioni generali su AWS
Configura la regione AWS, i cloud privati virtuali (VPCs), gli intervalli CIDR, le zone di disponibilità e le sottoreti.		Informazioni generali su AWS

Operazione	Description	Competenze richieste
Configura l'istanza di replica AWS DMS.	L'istanza di replica AWS DMS può connettersi sia ai database locali che a quelli AWS.	Informazioni generali su AWS
Configura gli endpoint AWS DMS.	Configura gli endpoint per i database di origine e di destinazione.	Informazioni generali su AWS

Prepara i fogli di lavoro per le attività e i tag di AWS DMS

Operazione	Description	Competenze richieste
Configura l'elenco delle tabelle.	Elenca tutte le tabelle coinvolte nella migrazione.	Database
Prepara il foglio di lavoro delle attività.	Prepara il foglio di lavoro di Excel utilizzando l'elenco delle tabelle che hai configurato.	Informazioni generali su AWS, Microsoft Excel
Prepara il foglio di lavoro per i tag.	Dettagli i tag delle risorse AWS da allegare alle attività di AWS DMS.	Informazioni generali su AWS, Microsoft Excel

Scarica ed esegui lo strumento

Operazione	Description	Competenze richieste
Scarica ed estrai lo strumento di generazione dei modelli dal GitHub repository.	GitHub archivio:/ https://github.com/aws-samples/dms-cloudformation-templates-generator	
Esegui lo strumento.	Segui le istruzioni dettagliate nel post del blog elencato	

Operazione	Description	Competenze richieste
	nella sezione «Riferimenti e aiuto».	

Risorse correlate

- Crea CloudFormation modelli AWS per attività AWS DMS utilizzando Microsoft Excel (post sul blog)
- Generatore di CloudFormation modelli DMS (repository) GitHub
- Documentazione in Python
- descrizione e download in formato xlrd
- Documentazione AWS DMS
- CloudFormation Documentazione AWS

Inizia con l'individuazione automatica del portafoglio

Pratik Chunawala e Rodolfo Jr. Cerrada, Amazon Web Services

Riepilogo

La valutazione del portafoglio e la raccolta dei metadati rappresentano una sfida fondamentale durante la migrazione di applicazioni e server su Amazon Web Services (AWS) Cloud, in particolare per le migrazioni di grandi dimensioni con più di 300 server. L'utilizzo di uno strumento di rilevamento automatico del portafoglio può aiutarti a raccogliere informazioni sulle tue applicazioni, come il numero di utenti, la frequenza di utilizzo, le dipendenze e le informazioni sull'infrastruttura dell'applicazione. Queste informazioni sono essenziali per pianificare le ondate di migrazione, in modo da poter assegnare priorità e raggruppare correttamente le applicazioni con caratteristiche simili. L'utilizzo di uno strumento di rilevamento semplifica la comunicazione tra il team del portfolio e i proprietari delle applicazioni, poiché il team del portfolio può convalidare i risultati dello strumento di rilevamento anziché raccogliere manualmente i metadati. Questo modello illustra le considerazioni chiave per la scelta di uno strumento di rilevamento automatizzato e le informazioni su come implementarne e testarne uno nel proprio ambiente.

Questo modello include un modello, che è un punto di partenza per creare la propria lista di controllo di attività di alto livello. Accanto alla lista di controllo c'è un modello per una matrice RACI (responsabile, responsabile, consultata, informata). È possibile utilizzare questa matrice RACI per determinare chi è responsabile di ogni attività inclusa nella lista di controllo.

Epiche

Seleziona uno strumento di scoperta

Operazione	Description	Competenze richieste
Determina se uno strumento di rilevamento è appropriato per il tuo caso d'uso.	Uno strumento di scoperta potrebbe non essere la soluzione migliore per il tuo caso d'uso. Considerate la quantità di tempo necessari a per selezionare, procurare , preparare e implementare uno strumento di scoperta.	Responsabile della migrazion e, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	Possono essere necessari e dalle 4 alle 8 settimane per configurare l'appliance di scansione per uno strumento di rilevamento senza agenti nell'ambiente in uso o per installare gli agenti per tutti i carichi di lavoro pertinent i. Una volta implementato, è necessario attendere 4-12 settimane prima che lo strumento di rilevamento raccolga i metadati mediante la scansione dei carichi di lavoro delle applicazi oni e l'analisi dello stack delle applicazioni. Se state migrando meno di 100 server, potreste essere in grado di raccogliere manualmente i metadati e analizzare le dipendenze più velocemen te del tempo necessario per distribuire e raccogliere i metadati con uno strumento di rilevamento automatico.	

Operazione	Description	Competenze richieste
Seleziona uno strumento di scoperta.	Consulta le considera zioni sulla selezione di uno strumento di rilevamento automatico nella sezione Informazioni aggiuntive. Determina i criteri appropria ti per la selezione di uno strumento di rilevamento per il tuo caso d'uso, quindi valuta ogni strumento in base a tali criteri. Per un elenco completo degli strumenti di rilevamento automatiz zato, consulta Strumenti di migrazione Discovery, Planning e Recommendation.	Responsabile della migrazion e, ingegnere addetto alla migrazione

Prepararsi per l'installazione

Operazione	Description	Competenze richieste
Preparare la lista di controllo prima dell'implementazione.	Crea una lista di controllo delle attività da completare prima di distribuire lo strumento . Per un esempio, consulta Predeployment Checklist sul sito Web della documenta zione di Flexera.	Responsabile della costruzio ne, ingegnere addetto alla migrazione, responsabile della migrazione, amministratore di rete
Prepara i requisiti di rete.	Fornisci le porte, i protocoll i, gli indirizzi IP e il routing necessari per l'esecuzione dello strumento e l'accesso ai server di destinazione. Per	Ingegnere addetto alla migrazione, amministratore di rete, architetto cloud

Operazione	Description	Competenze richieste
	ulteriori informazioni, consulta la guida all'installazione del tuo strumento di rilevamento. Per un esempio, consulta Requisiti di implementazione sul sito Web della documentazione di Flexera.	
Prepara i requisiti relativi all'account e alle credenziali.	Identifica le credenziali necessarie per accedere ai server di destinazione e installare tutti i componenti dello strumento.	Amministratore cloud, General AWS, ingegnere addetto alla migrazione, responsabile della migrazione, amministratore di rete, amministratore AWS
Prepara le appliance su cui installerai lo strumento.	Assicuratevi che i dispositivi su cui installerete i componenti dell'utensile soddisfino le specifiche e i requisiti di piattaforma dello strumento.	Ingegnere addetto alla migrazione, responsabile della migrazione, amministratore di rete
Prepara gli ordini di modifica.	In base al processo di gestione delle modifiche in atto nell'organizzazione, preparate gli eventuali ordini di modifica necessari e assicuratevi che tali ordini di modifica siano approvati.	Costruisci lead, migrazioni leader
Invia i requisiti alle parti interessate.	Invia la checklist di pre-implementazione e i requisiti di rete alle parti interessate. Le parti interessate devono esaminare, valutare e preparare i requisiti necessari prima di procedere con l'implementazione.	Crea un vantaggio, guida la migrazione

Implementa lo strumento

Operazione	Description	Competenze richieste
Scarica il programma di installazione.	Scarica il programma di installazione o l'immagine della macchina virtuale. Le immagini delle macchine virtuali sono generalmente disponibili in formato Open Virtualization Format (OVF).	Crea un lead, un responsabile della migrazione
Estrai i file.	Se si utilizza un programma di installazione, è necessari o scaricare ed eseguire il programma di installazione su un server locale.	Crea un lead, un responsabile della migrazione
Implementa lo strumento sui server.	Implementa lo strumento di rilevamento sui server locali di destinazione come segue: • Se il file di origine è un'immagine di macchina virtuale, distribuiscilo nell'ambiente della macchina virtuale, ad esempio. VMware • Se il file sorgente è un programma di installazione, esegui il programma di installazione per installare e configurare lo strumento.	Responsabile build, responsab ile della migrazione, amministr atore di rete
Accedi allo strumento di scoperta.	Segui le istruzioni sullo schermo e accedi per iniziare a usare lo strumento.	Responsabile della migrazion e, crea lead

Operazione	Description	Competenze richieste
Attiva il prodotto.	Inserisci la tua chiave di licenza.	Crea un lead, un responsabile della migrazione
Configurare lo strumento.	Immettere le credenziali necessarie per accedere ai server di destinazione, ad esempio credenziali per Windows VMware, Simple Network Management Protocol (SNMP) e Secure Shell Protocol (SSH) o database.	Crea lead, responsabile della migrazione

Prova lo strumento

Operazione	Description	Competenze richieste
Seleziona i server di test.	Identifica un piccolo set di sottoreti o indirizzi IP non di produzione da utilizzare per testare lo strumento di rilevamento. Ciò consente di convalidare rapidamen te le scansioni, identificare e risolvere rapidamente eventuali errori e isolare i test dagli ambienti di produzione.	Crea lead, responsabile della migrazione, amministratore di rete
Inizia la scansione dei server di test selezionati.	Per uno strumento di rilevamento senza agente, inserisci le sottoreti o gli indirizzi IP per i server di test selezionati nella console dello strumento di rilevamento e avvia la scansione.	Responsabile build, responsab ile della migrazione, amministr atore di rete

Operazione	Description	Competenze richieste
	Per uno strumento di rilevamento basato su agenti, installa l'agente sui server di test selezionati.	
Esamina i risultati della scansione.	Esamina i risultati della scansione per i server di test. Se vengono rilevati errori, risolvili e correggili. Documenta gli errori e le soluzioni. Utilizzerai queste informazioni in futuro e potrai aggiungerle al tuo portfolio runbook.	Responsabile build, responsabile della migrazione, amministratore di rete
Scansiona nuovamente i server di test.	Una volta completata la nuova scansione, ripeti la scansione fino a eliminare gli errori.	Responsabile build, responsabile della migrazione, amministratore di rete

Risorse correlate

AWS resources

- Guida alla valutazione del portafoglio di applicazioni per la Cloud AWS migrazione
- Strumenti di migrazione Discovery, Planning e Recommendation

Guide all'implementazione per strumenti di rilevamento comunemente selezionati

- Implementa l'appliance virtuale RN15 0 (documentazione Flexera)
- Installazione di Gatherer (documentazione ModelizeIT)
- Installazione On-Prem Analysis Server (documentazione ModelizeIT)

Informazioni aggiuntive

Considerazioni sulla scelta di uno strumento di rilevamento automatico

Ogni strumento di scoperta presenta vantaggi e limiti. Quando selezionate lo strumento appropriato per il vostro caso d'uso, tenete presente quanto segue:

- Scegliete uno strumento di scoperta in grado di raccogliere la maggior parte, se non tutti, i metadati necessari per raggiungere l'obiettivo di valutazione del portafoglio.
- Identifica i metadati che devi raccogliere manualmente perché lo strumento non li supporta.
- Fornisci i requisiti dello strumento di rilevamento alle parti interessate in modo che possano esaminarlo e valutarlo in base ai requisiti interni di sicurezza e conformità, come i requisiti di server, rete e credenziali.
  - Lo strumento richiede l'installazione di un agente nel carico di lavoro pertinente?
  - Lo strumento richiede la configurazione di un'appliance virtuale nel proprio ambiente?
- Determina i requisiti di residenza dei dati. Alcune organizzazioni non vogliono archiviare i propri dati al di fuori del proprio ambiente. Per risolvere questo problema, potrebbe essere necessario installare alcuni componenti dello strumento nell'ambiente locale.
- Assicurati che lo strumento supporti il sistema operativo (OS) e la versione del sistema operativo del carico di lavoro pertinente.
- Determina se il tuo portafoglio include server mainframe, di fascia media e legacy. La maggior parte degli strumenti di rilevamento è in grado di rilevare questi carichi di lavoro come dipendenze, ma alcuni strumenti potrebbero non essere in grado di ottenere dettagli sul dispositivo, come l'utilizzo e le dipendenze del server. Gli strumenti di rilevamento Device42 e ModernizeIT supportano entrambi server mainframe e di fascia media.

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Esegui la migrazione dei carichi di lavoro Cloudera locali a Cloudera Data Platform su AWS

Battulga Purevragchaa e Nidhi Gupta, Amazon Web Services

Nijjwol Lamsal, Cloudera, Inc.

Riepilogo

Questo modello descrive i passaggi di alto livello per la migrazione dei carichi di lavoro Cloudera Distributed Hadoop (CDH), Hortonworks Data Platform (HDP) e Cloudera Data Platform (CDP) locali su CDP Public Cloud su AWS. Ti consigliamo di collaborare con Cloudera Professional Services e un integratore di sistemi (SI) per implementare questi passaggi.

Ci sono molte ragioni per cui i clienti Cloudera vogliono spostare i carichi di lavoro CDH, HDP e CDP locali sul cloud. Alcuni motivi tipici includono:

- Semplifica l'adozione di nuovi paradigmi di piattaforme dati come data lakehouse o data mesh
- Aumenta l'agilità aziendale, democratizza l'accesso e l'inferenza sugli asset di dati esistenti
- Riduci il costo totale di proprietà (TCO)
- Migliora l'elasticità del carico di lavoro
- Consenti una maggiore scalabilità; riduci drasticamente i tempi di fornitura dei servizi dati rispetto alla base di installazioni legacy in locale
- Ritirate l'hardware obsoleto; riducete in modo significativo i cicli di aggiornamento dell'hardware
- Sfrutta pay-as-you-go i prezzi, che sono estesi ai carichi di lavoro Cloudera su AWS con il modello di licenza Cloudera (CCU)
- Sfrutta i vantaggi di una distribuzione più rapida e di una migliore integrazione con piattaforme di integrazione continua e distribuzione continua (CI/CD)
- Utilizza un'unica piattaforma unificata (CDP) per più carichi di lavoro

Cloudera supporta tutti i principali carichi di lavoro, tra cui Machine Learning, Data Engineering, Data Warehouse, Operational Database, Stream Processing (CSP) e sicurezza e governance dei dati. Cloudera offre questi carichi di lavoro in locale da molti anni e puoi migrarli sul cloud AWS utilizzando CDP Public Cloud con Workload Manager e Replication Manager.

Cloudera Shared Data Experience (SDX) fornisce un catalogo di metadati condiviso tra questi carichi di lavoro per facilitare la gestione e le operazioni coerenti dei dati. SDX include anche una sicurezza

completa e granulare per la protezione dalle minacce e una governance unificata per funzionalità di audit e ricerca per la conformità a standard come Payment Card Industry Data Security Standard (PCI DSS) e GDPR.

La migrazione CDP a colpo d'occhio

Carico di lavoro	Carico di lavoro di origine	CDH, HDP e CDP Private Cloud
	Ambiente di origine	• Windows, Linux • In locale, in colocation o in qualsiasi ambiente non AWS
	Carico di lavoro di destinazione	Cloud pubblico CDP su AWS
	Ambiente di destinazione	• Modello di implementazione: account cliente • Modello operativo: piano customer/Cloudera di controllo
	Strategia di migrazione (7R)	Rehost, ripiattaforma o refactor
Migrazione	Si tratta di un aggiornamento della versione Workload?	Sì
	Durata della migrazione	• Implementazione: circa 1 settimana per creare un account cliente, un cloud privato virtuale (VPC) e un ambiente gestito dai clienti CDP Public Cloud. • Durata della migrazione: 1-4 mesi, a seconda della complessità e delle

dimensioni del carico di lavoro.

Costo	Costo di esecuzione del carico di lavoro su AWS	• Ad un livello elevato, il costo di una migrazione del carico di lavoro CDH verso AWS presuppone la creazione di un nuovo ambiente su AWS. Include la contabilizzazione del tempo e dell'impegno del personale, nonché la fornitura di risorse informatiche e la concessione di licenze software per il nuovo ambiente. • Il modello di prezzo basato sul consumo del cloud di Cloudera ti offre la flessibilità necessaria per sfruttare le funzionalità di espansione e di scalabilità automatica. Per ulteriori informazioni, consulta le tariffe dei servizi CDP Public Cloud sul sito Web di Cloudera. • Cloudera Enterprise Data Hub si basa su Amazon Elastic Compute Cloud EC2 (Amazon) e modella fedelmente i cluster tradizionali. Data Hub può essere personalizzato, ma ciò influirà sui costi. • CDP Public Cloud Data Warehouse, Cloudera Machine Learning e Cloudera Data Engineering (CDE) sono basati su contenitori e possono

		essere configurati per scalare automaticamente.
Accordi e quadro di infrastruttura	Requisiti di sistema	Vedi la sezione Prerequisiti.
	SLA	Consulta l'accordo sul livello di servizio di Cloudera per CDP Public Cloud.
	DOTT.	Vedi Disaster Recovery nella documentazione di Cloudera.
	Modello operativo e di licenza (per l'account AWS di destinazione)	Modello Bring Your Own License (BYOL)
Conformità	Requisiti in materia di sicurezza	Consulta la panoramica sulla sicurezza di Cloudera nella documentazione di Cloudera.
	Altre certificazioni di conformità	Consulta le informazioni sul sito web di Cloudera sulla conformità al Regolamento generale sulla protezione dei dati (GDPR) e sul CDP Trust Center.

Prerequisiti e limitazioni

Prerequisiti

- Requisiti dell'account AWS, inclusi account, risorse, servizi e autorizzazioni, come la configurazione di ruoli e policy di AWS Identity and Access Management (IAM)
- Prerequisiti per la distribuzione di CDP dal sito Web di Cloudera

La migrazione richiede i seguenti ruoli e competenze:

Ruolo	Competenze e responsabilità
Responsabile della migrazione	Garantisce il supporto esecutivo, la collaborazione tra i team, la pianificazione, l'implementazione e la valutazione
Cloudera SME	Competenze specialistiche in amministrazione, amministrazione di sistema e architettura CDH, HDP e CDP
Architetto AWS	Competenze nei servizi, nelle reti, nella sicurezza e nelle architetture AWS

Architecture

Utilizzare l'architettura appropriata è un passaggio fondamentale per garantire che la migrazione e le prestazioni soddisfino le aspettative. Affinché la migrazione soddisfi i presupposti di questo playbook, l'ambiente di dati di destinazione nel cloud AWS, su istanze ospitate su cloud privato virtuale (VPC) o CDP, deve corrispondere all'ambiente di origine in termini di sistema operativo e versioni del software, nonché delle principali specifiche delle macchine.

Il diagramma seguente (riprodotto con l'autorizzazione della scheda tecnica di Cloudera Shared Data Experience) mostra i componenti dell'infrastruttura per l'ambiente CDP e come interagiscono i livelli o i componenti dell'infrastruttura.

L'architettura include i seguenti componenti CDP:

- Data Hub è un servizio per l'avvio e la gestione di cluster di carichi di lavoro basato su Cloudera Runtime. Puoi utilizzare le definizioni dei cluster in Data Hub per fornire e accedere ai cluster di carichi di lavoro per casi d'uso personalizzati e definire configurazioni di cluster personalizzate. Per ulteriori informazioni, consulta il sito Web di Cloudera.
- Data Flow and Streaming affronta le principali sfide che le aziende devono affrontare con i dati in movimento. Gestisce quanto segue:
  - Elaborazione di flussi di dati in tempo reale ad alto volume e su larga scala
  - Monitoraggio della provenienza dei dati e della provenienza dei dati in streaming
  - Gestione e monitoraggio delle applicazioni periferiche e delle fonti di streaming

Per ulteriori informazioni, consulta Cloudera DataFlow e CSP sul sito Web di Cloudera.

- L'ingegneria dei dati include l'integrazione dei dati, la qualità dei dati e la governance dei dati, che aiutano le organizzazioni a creare e mantenere pipeline e flussi di lavoro di dati. Per ulteriori informazioni, consulta il sito Web di Cloudera. Scopri il supporto per le istanze spot per facilitare il risparmio sui costi sui carichi di lavoro AWS for Cloudera Data Engineering.
- Data Warehouse ti consente di creare data warehouse e data mart indipendenti che si ridimensionano automaticamente per soddisfare le richieste di carico di lavoro. Questo servizio fornisce istanze di elaborazione isolate e ottimizzazione automatizzata per ogni data warehouse e data mart e consente di ridurre i costi durante le riunioni. SLAs Per ulteriori informazioni, consulta il sito Web di Cloudera. Scopri come gestire i costi e l'auto-scaling per Cloudera Data Warehouse on AWS.
- Il database operativo in CDP fornisce una base affidabile e flessibile per applicazioni scalabili e ad alte prestazioni. Fornisce un database scalabile in tempo reale, sempre disponibile e che serve dati strutturati tradizionali insieme a nuovi dati non strutturati all'interno di una piattaforma operativa e di magazzino unificata. Per ulteriori informazioni, consulta il sito Web di Cloudera.
- Machine Learning è una piattaforma di machine learning nativa per il cloud che unisce funzionalità self-service di data science e ingegneria dei dati in un unico servizio portatile all'interno di un cloud di dati aziendale. Consente l'implementazione scalabile dell'apprendimento automatico e dell'intelligenza artificiale (AI) sui dati ovunque. Per ulteriori informazioni, consulta il sito Web di Cloudera.

CDP su AWS

Il diagramma seguente (adattato con l'autorizzazione del sito Web di Cloudera) mostra l'architettura di alto livello di CDP su AWS. CDP implementa il proprio modello di sicurezza per gestire sia gli account che il flusso di dati. Questi sono integrati con IAM tramite l'uso di ruoli tra account.

Il piano di controllo CDP risiede in un account master Cloudera nel proprio VPC. Ogni account cliente ha il proprio account secondario e un VPC unico. I ruoli IAM e le tecnologie SSL su più account indirizzano il traffico di gestione da e verso il piano di controllo ai servizi clienti che risiedono su sottoreti pubbliche instradabili su Internet all'interno del VPC di ciascun cliente. Sul VPC del cliente, Cloudera Shared Data Experience (SDX) offre una sicurezza di livello aziendale con governance e conformità unificate in modo da poter ottenere informazioni dai dati più velocemente. SDX è una filosofia di progettazione incorporata in tutti i prodotti Cloudera. Per ulteriori informazioni su SDX e l'architettura di rete CDP Public Cloud per AWS, consulta la documentazione di Cloudera.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon Elastic Kubernetes Service (Amazon EKS) ti aiuta a eseguire Kubernetes su AWS senza dover installare o gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Automazione e strumenti

- Per strumenti aggiuntivi, puoi utilizzare Cloudera Backup Data Recovery (BDR), AWS Snowball e AWS Snowmobile per facilitare la migrazione dei dati da CDH, HDP e CDP locali a CDP ospitati da AWS.
- Per le nuove implementazioni, ti consigliamo di utilizzare la soluzione AWS Partner per CDP.

Epiche

Preparati per la migrazione

Operazione	Description	Competenze richieste
Coinvolgi il team di Cloudera.	Cloudera persegue un modello di coinvolgimento standardizzato con i propri clienti e può collaborare con il vostro integratore di sistemi (SI) per promuovere lo stesso approccio. Contatta il team clienti di Cloudera in modo	Responsabile della migrazione

Operazione	Description	Competenze richieste
	che possa fornire indicazio ni e le risorse tecniche necessarie per avviare il progetto. Contattando il team di Cloudera si garantisce che tutti i team necessari possano prepararsi per la migrazion e all'avvicinarsi della data prevista. Puoi contattare Cloudera Professional Services per spostare l'implementazione di Cloudera dalla fase pilota a quella di produzione in modo rapido, a costi inferiori e con prestazioni ottimali. Per un elenco completo delle offerte, consulta il sito Web di Cloudera.	
Crea un ambiente CDP Public Cloud su AWS per il tuo VPC.	Collabora con Cloudera Professional Services o il tuo SI per pianificare e distribuire CDP Public Cloud in un VPC su AWS.	Architetto del cloud, Cloudera SME

Operazione	Description	Competenze richieste
Assegna priorità e valuta i carichi di lavoro per la migrazione.	Valuta tutti i carichi di lavoro locali per determinare i carichi di lavoro più facili da migrare. Le applicazioni che non sono mission-critical sono le migliori da spostare per prime, perché avranno un impatto minimo sui clienti. Salva i carichi di lavoro mission-critical per ultimi, dopo aver migrato con successo altri carichi di lavoro. **Note** I carichi di lavoro transitori (CDP Data Engineering) sono più facili da migrare rispetto ai carichi di lavoro persistenti (CDP Data Warehouse). È inoltre importante considerare il volume e le posizioni dei dati durante la migrazion e. Le sfide possono includere la replica continua dei dati da un ambiente locale al cloud e la modifica delle pipeline di inserimento dei dati	Responsabile della migrazione

Operazione	Description	Competenze richieste
	per importare i dati direttamente nel cloud.	
Discutete delle attività di migrazione di CDH, HDP, CDP e applicazioni legacy.	Prendi in considerazione e inizia a pianificare le seguenti attività con Cloudera Workload Manager: • Dati e carichi di lavoro da copiare nel tuo ambiente AWS • Dati pronti per il cloud • Vicini rumorosi, che consumano risorse e creano problemi agli altri inquilini • Carichi di lavoro elastici • Cluster di piccole dimension i con sovraccarico operativo elevato	Responsabile della migrazione

Operazione	Description	Competenze richieste
Completa i requisiti e i consigli di Cloudera Replication Manager.	Collabora con Cloudera Professional Services e il tuo SI per prepararti a migrare i carichi di lavoro nel tuo ambiente CDP Public Cloud su AWS. La comprensione dei seguenti requisiti e consigli può aiutare a evitare problemi comuni durante e dopo l'installazione del servizio Replication Manager. • Consultate i documenti di supporto di Replication Manager per confermare che i requisiti di ambiente e sistema siano soddisfatti. Per ulteriori informazioni, consulta Support matrix for CDP Public Cloud Replication Manager sul sito Web di Cloudera. • Non è necessario l'accesso root ai nodi su cui verranno installati l'app Replication Manager e il motore Data Lifecycle Manager (DLM). • Installa Apache Hive durante l'installazione iniziale di Replication Manager, a meno che tu non sia certo che non utilizzerai la replica Hive in futuro. Se si decide	Responsabile della migrazione

Operazione	Description	Competenze richieste
	di installare Hive dopo aver creato le politiche di replica HDFS in Replicati on Manager, è necessario eliminare e quindi ricreare tutte le politiche di replica HDFS dopo aver aggiunto Hive. • I cluster utilizzati in Replicati on Manager devono avere configurazioni simmetriche. Ogni cluster in una relazione di replica deve essere configurato esattamen te allo stesso modo per la sicurezza (Kerberos), la gestione degli utenti (LDAP/AD) e il proxy Knox. I servizi cluster come Hadoop Distributed File System (HDFS), Apache Hive, Apache Knox, Apache Ranger e Apache Atlas possono avere configura zioni diverse per l'alta disponibilità (HA). Ad esempio, i cluster di origine e di destinazione potrebber o avere configurazioni HA e non HA separate.	

Esegui la migrazione da CDP ad AWS

Operazione	Description	Competenze richieste
Esegui la migrazione del primo carico di lavoro per dev/test ambienti utilizzando Cloudera Workload Manager.	Il tuo SI può aiutarti a migrare il tuo primo carico di lavoro nel cloud AWS. Questa dovrebbe essere un'applicazione che non sia rivolta ai clienti o che non sia fondamentale per la missione. I candidati ideali per la dev/test migrazione sono le applicazioni che contengono dati che il cloud può facilment e importare, come i carichi di lavoro CDP Data Engineering. Si tratta di un carico di lavoro temporaneo a cui in genere accedono meno utenti, rispetto a un carico di lavoro persisten te come un carico di lavoro CDP Data Warehouse, che potrebbe avere molti utenti che necessitano di un accesso ininterrotto. I carichi di lavoro di data engineering non sono persistenti, il che riduce al minimo l'impatto aziendale in caso di problemi. Tuttavia, questi lavori potrebbero essere fondamentali per i report di produzione, quindi dai la priorità ai carichi di lavoro di Data Engineering a basso impatto.	Responsabile della migrazione

Operazione	Description	Competenze richieste
Ripetere i passaggi di migrazione se necessario.	Cloudera Workload Manager aiuta a identificare i carichi di lavoro più adatti per il cloud. Fornisce metriche come valutazioni delle prestazioni del cloud, sizing/capacity piani per l'ambiente di destinazi one e piani di replica. I migliori candidati per la migrazion e sono i carichi di lavoro stagionali, i report ad hoc e i lavori intermittenti che non consumano molte risorse. Cloudera Replication Manager sposta i dati dall'ambiente locale al cloud e dal cloud all'ambiente locale. Ottimizza in modo proattivo carichi di lavoro, applicazi oni, prestazioni e capacità dell'infrastruttura per il data warehousing, l'ingegneria dei dati e l'apprendimento automatico utilizzando Workload Manager. Per una guida completa su come modernizzare un data warehouse, consulta il sito Web di Cloudera.	Cloudera PMI

Risorse correlate

Documentazione Cloudera:

- Registrazione di cluster classici con CDP, Cloudera Manager e Replication Manager:
  - Console di gestione
  - Replica dell'hive di Replication Manager
- Replica Sentry
- Autorizzazioni Sentry
- Lista di controllo per la pianificazione dei cluster Data Hub
- Architettura di Workload Manager
- Requisiti di Replication Manager
- Osservabilità della piattaforma dati di Cloudera
- Requisiti AWS

Documentazione AWS:

- Migrazione dei dati del cloud

Risolvi gli errori di connessione dopo la migrazione di Microsoft SQL Server al cloud AWS

Premkumar Chelladurai, Amazon Web Services

Riepilogo

Dopo aver migrato Microsoft SQL Server in esecuzione su Windows Server 2008 R2, 2012 o 2012 R2 su istanze Amazon Elastic Compute Cloud (Amazon EC2) sul cloud Amazon Web Services (AWS), la connessione a SQL Server non riesce e vengono visualizzati i seguenti errori:

- `[Microsoft][ODBC SQL Server Driver][DBNETLIB] General Network error`
- `ERROR [08S01] [Microsoft][SQL Native Client]Communication link failure. System.Data.SqlClient.SqlException: A transport-level error has occurred when sending the request to the server. (provider: TCP Provider, error: 0 - An existing connection was forcibly closed by the remote host.)`
- `TCP Provider: The semaphore timeout period has expired`

Questo modello descrive come risolvere questi errori disattivando le funzionalità di Windows Scalable Networking Pack (SNP) a livello di sistema operativo (OS) e interfaccia di rete per SQL Server in esecuzione su Windows Server 2008 R2, 2012 o 2012 R2.

Prerequisiti e limitazioni

Prerequisiti

- Privilegi di amministratore per Windows Server.
- Se hai utilizzato AWS Application Migration Service come strumento di migrazione, hai bisogno di una delle seguenti versioni di Windows Server:
  - Windows Server 2008 R2 Service Pack 1, 2012 o 2012 R2
- Se hai utilizzato CloudEndure Migration come strumento di migrazione, hai bisogno di una delle seguenti versioni di Windows Server:
  - Windows Server 2003 R2 Service Pack 3, 2008, 2008 R2 Service Pack 1, 2012 o 2012 R2

Tools (Strumenti)

- Amazon EC2 — Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi usare Amazon EC2 per lanciare tutti o pochi server virtuali di cui hai bisogno, con scalabilità orizzontale o orizzontale.
- Windows Server: Windows Server è una piattaforma per la creazione di un'infrastruttura di applicazioni, reti e servizi Web connessi.

Epiche

Disattiva le funzionalità SNP a livello di sistema operativo e elastic network interface

Operazione	Description	Competenze richieste
Disattiva le funzionalità SNP a livello di sistema operativo.	1. Accedi a Windows Server e apri un prompt dei comandi come amministratore. 2. Esegui il comando `netsh int tcp show global`. 3. Nell'output, controlla se uno dei due `Receive-Side Scaling` `Chimney Offload` è in `enabled` modalità. Se uno dei due lo è`enabled`, esegui i seguenti comandi: • `netsh int tcp set global chimney=disabled` • `netsh int tcp set global rss=disabled`	Amministratore AWS, amministratore di sistema AWS, ingegnere addetto alla migrazione, amministratore cloud
Disattiva le funzionalità SNP a livello di elastic network interface.	1. Scegliete Start, immettete `ncpa.cpl`, quindi premete Invio.	Amministratore AWS, amministratore cloud,

Operazione	Description	Competenze richieste
	2. Fai clic con il pulsante destro del mouse su Elastic 3. Nel menu popup, scegli Proprietà. 4. Nella finestra Proprietà dell'adattatore Ethernet, scegli Configura. 5. Nella finestra popup Amazon Elastic Network Adapter Properties, scegli la scheda Advanced. 6. Nella sezione Proprietà, disattiva tutti gli offload e gli RSS.	amministratore di sistema AWS

Risorse correlate

- Come risolvere i problemi relativi a funzionalità avanzate di prestazioni di rete come RSS e NetDMA

Riavvia automaticamente l'AWS Replication Agent senza disabilitarlo SELinux dopo il riavvio di un server di origine RHEL

Anil Kunapareddy, Venkatramana Chintha e Shanmugam Shanker, Amazon Web Services

Riepilogo

AWS Application Migration Service aiuta a semplificare, accelerare e automatizzare la migrazione del carico di lavoro Red Hat Enterprise Linux (RHEL) al cloud Amazon Web Services (AWS). Per aggiungere server di origine a Application Migration Service, installa l'AWS Replication Agent sui server.

Application Migration Service fornisce una replica asincrona a livello di blocco in tempo reale. Ciò significa che è possibile continuare le normali operazioni IT durante l'intero processo di replica. Queste operazioni IT potrebbero richiedere il riavvio o il riavvio del server di origine RHEL durante la migrazione. In tal caso, l'agente di replica AWS non si riavvierà automaticamente e la replica dei dati si interromperà. In genere, puoi impostare Security-Enhanced Linux (SELinux) in modalità disabilitata o permissiva per riavviare automaticamente AWS Replication Agent. Tuttavia, le politiche di sicurezza della tua organizzazione potrebbero proibire la disabilitazione e potresti anche dover SELinux rietichettare i file.

Questo modello descrive come riavviare automaticamente AWS Replication Agent senza spegnerlo SELinux quando il server di origine RHEL si riavvia o si riavvia durante una migrazione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un carico di lavoro RHEL locale che desideri migrare nel cloud AWS.
- Application Migration Service inizializzato dalla console di Application Migration Service. L'inizializzazione è richiesta solo la prima volta che si utilizza questo servizio. Per istruzioni, consulta la documentazione di Application Migration Service.
- Una policy AWS Identity and Access Management (IAM) esistente per Application Migration Service. Per ulteriori informazioni, consulta la documentazione di Application Migration Service.

Versioni

- RHEL versione 7 o successiva

Tools (Strumenti)

Servizi AWS

- AWS Application Migration Service è una soluzione altamente automatizzata lift-and-shift (rehosting) che semplifica, accelera e riduce i costi di migrazione delle applicazioni su AWS.

Comandi Linux

La tabella seguente fornisce un elenco di comandi Linux che verranno eseguiti sul server di origine RHEL. Questi sono descritti anche nei poemi epici e nelle storie di questo modello.

Comando	Description
`#systemctl –version`	Identifica la versione del sistema.
`#systemctl list-units --type=service`	Elenca tutti i servizi attivi disponibili sul server RHEL.
`#systemctl list-units --type=service | grep running`	Elenca tutti i servizi attualmente in esecuzione sul server RHEL.
`#systemctl list-units --type=service | grep failed`	Elenca tutti i servizi che non sono stati caricati dopo il riavvio o il riavvio del server RHEL.
`restorecon -Rv /etc/rc.d/init.d/aws-replication-service`	Modifica il contesto in. `aws-replication-service`
`yum install policycoreutils*`	Installa le utilità principali della policy necessari e per il funzionamento del SELinux sistema.
`ausearch -c "insmod" --raw | audit2allow -M my-modprobe`	Cerca nel registro di controllo e crea un modulo per le politiche.
`semodule -i my-modprobe.pp`	Attiva la politica.

`cat my-modprobe.te`	Visualizza il contenuto del `my-modprobe.te` file.
`semodule -l | grep my-modprobe`	Verifica se la politica è stata caricata nel SELinux modulo.

Epiche

Installa AWS Replication Agent e riavvia il server di origine RHEL

Operazione	Description	Competenze richieste
Crea un utente di Application Migration Service con una chiave di accesso e una chiave di accesso segreta.	Per installare AWS Replication Agent, devi creare un utente di Application Migration Service con le credenziali AWS richieste. Per istruzioni, consulta la documentazione di Application Migration Service.	Tecnico di migrazione
Installa l'agente di replica AWS.	1. Accedi alla Console di gestione AWS e apri la console AWS Migration Service da https://console.aws.amazon.com/mgn/casa. 2. Configura le impostazioni di replica seguendo le istruzioni nella documentazione di Application Migration Service. 3. Installa AWS Replication Agent seguendo le istruzioni nella documentazione di Application Migration Service.	Tecnico di migrazione

Operazione	Description	Competenze richieste
	4. Nella pagina Server di origine, scegli il server di origine RHEL, quindi scegli Replica per avviare la replica iniziale. Per ulteriori informazioni, consulta la documentazione di Application Migration Service.	
Riavvia o riavvia il server di origine RHEL.	Riavvia o riavvia il server di origine RHEL quando lo stato di replica dei dati viene visualizzato Stallato nel dashboard di migrazione.	Ingegnere della migrazione
Controlla lo stato della replica dei dati.	Attendi un'ora, quindi controlla nuovamente lo stato della replica dei dati nella dashboard di migrazione. Dovrebbe essere nello stato Integro.	Ingegnere della migrazione

Verifica lo stato di AWS Replication Agent sul server di origine RHEL

Operazione	Description	Competenze richieste
Identifica la versione del sistema.	Apri l'interfaccia a riga di comando per il tuo server di origine RHEL ed esegui il seguente comando per identificare la versione del sistema: `#systemctl –version`	Ingegnere della migrazione

Operazione	Description	Competenze richieste
Elenca tutti i servizi attivi.	Per elencare tutti i servizi attivi disponibili sul server RHEL, esegui il comando: `#systemctl list-units --type=service`	Ingegnere della migrazione
Elenca tutti i servizi in esecuzione.	Per elencare tutti i servizi attualmente in esecuzione sul server RHEL, utilizzare il comando: `#systemctl list-units --type=service | grep running`	Ingegnere della migrazione
Elenca tutti i servizi che non sono stati caricati.	Per elencare tutti i servizi che non sono stati caricati dopo il riavvio o il riavvio del server RHEL, esegui il comando: `#systemctl list-units --type=service | grep failed`	Ingegnere della migrazione

Crea ed esegui il SELinux modulo

Operazione	Description	Competenze richieste
Cambia il contesto di sicurezza.	Nell'interfaccia a riga di comando per il tuo server di origine RHEL, esegui il seguente comando per modificare il contesto di	Ingegnere della migrazione

Operazione	Description	Competenze richieste
	sicurezza nel servizio di replica AWS: `restorecon -Rv /etc/rc.d/init.d/aws-replication-service`	
Installa le utilità principali.	Per installare le principali utilità necessarie per il funzionamento del SELinux sistema e le relative politiche, esegui il comando: `yum install policycoreutils*`	Ingegnere della migrazione
Cerca nel registro di controllo e crea un modulo per le politiche.	Eseguire il comando : `ausearch -c "insmod" --raw | audit2allow -M my-modprobe`	Ingegnere della migrazione
Visualizza il contenuto del my-modprobe-te file.	Il `my-modprobe.te` file viene generato dal comando audit2allow. Include i SELinux domini, la directory di origine delle politiche e le sottodirectory e specifica le regole e le transizioni dei vettori di accesso associate ai domini. Per visualizzare il contenuto del file, esegui il comando: `cat my modprobe.te`	Ingegnere della migrazione

Operazione	Description	Competenze richieste
Attiva la politica.	Per inserire il modulo e rendere attivo il pacchetto di policy, esegui il comando: `semodule -i my-modprobe.pp`	Ingegnere della migrazione
Controlla se il modulo è stato caricato.	Eseguire il comando : `semodule -l | grep my-modprobe` Dopo aver caricato il SELinux modulo, non sarà più necessario SELinux impostarlo sulla modalità disabilitata o permissiva durante la migrazione.	Ingegnere della migrazione
Riavviare o riavviare il server di origine RHEL e verificare lo stato della replica dei dati.	Apri la console AWS Migration Service, passa all'avanzamento della replica dei dati, quindi riavvia o riavvia il server di origine RHEL. La replica dei dati dovrebbe ora riprendere automaticamente dopo il riavvio del server di origine RHEL.	Ingegnere della migrazione

Risorse correlate

- Documentazione sul servizio di migrazione delle applicazioni
- Materiali per la formazione tecnica
- Risoluzione dei problemi di AWS Replication Agent
- Politiche del servizio di migrazione delle applicazioni

Re-architetto

Argomenti

- Converti VARCHAR2 (1) il tipo di dati per Oracle in tipo di dati booleano per Amazon Aurora PostgreSQL
- Crea utenti e ruoli delle applicazioni in Aurora, compatibile con PostgreSQL
- Emula Oracle DR utilizzando un database globale Aurora compatibile con PostgreSQL
- SHA1 Implementa l'hashing per i dati PII durante la migrazione da SQL Server a PostgreSQL
- Migrazione incrementale da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL utilizzando Oracle SQL Developer e AWS SCT
- Carica i file BLOB in formato TEXT utilizzando la codifica dei file in Aurora, compatibile con PostgreSQL
- Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL con e utilizzando e AWS SCT AWS DMS AWS CLI CloudFormation
- Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL in modalità SSL utilizzando AWS DMS
- Migrazione dei pacchetti pragma Oracle SERIALLY_REUSABLE in PostgreSQL
- Esegui la migrazione di tabelle esterne Oracle verso Amazon Aurora, compatibile con PostgreSQL
- Migrazione di indici basati su funzioni da Oracle a PostgreSQL
- Migrazione delle funzioni native di Oracle su PostgreSQL utilizzando le estensioni
- Esegui la migrazione di un database Db2 da Amazon a EC2 Aurora compatibile con MySQL utilizzando AWS DMS
- Esegui la migrazione di un database Microsoft SQL Server da Amazon EC2 ad Amazon DocumentDB utilizzando AWS DMS
- Esegui la migrazione di un database ThoughtSpot Falcon locale su Amazon Redshift
- Esegui la migrazione da Oracle Database ad Amazon RDS for PostgreSQL utilizzando Oracle GoldenGate
- Esegui la migrazione di una tabella partizionata Oracle su PostgreSQL utilizzando AWS DMS
- Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for MySQL
- Esegui la migrazione da IBM Db2 su Amazon a EC2 Aurora PostgreSQL compatibile con AWS DMS e AWS SCT

- Esegui la migrazione da Oracle 8i o 9i ad Amazon RDS for PostgreSQL utilizzando AWS DMS SharePlex
- Esegui la migrazione da Oracle 8i o 9i ad Amazon RDS for PostgreSQL utilizzando viste materializzate e AWS DMS
- Esegui la migrazione da Oracle su Amazon EC2 ad Amazon RDS for MySQL utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database Oracle da Amazon EC2 ad Amazon RDS for MariaDB utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for MySQL utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for PostgreSQL utilizzando un bystander Oracle e AWS DMS
- Esegui la migrazione di un database Oracle ad Amazon Redshift utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database Oracle ad Aurora PostgreSQL utilizzando AWS DMS e AWS SCT
- Esegui la migrazione dei dati da un database Oracle locale ad Aurora PostgreSQL
- Esegui la migrazione da SAP ASE ad Amazon RDS per SQL Server utilizzando AWS DMS
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon Redshift utilizzando AWS DMS
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon Redshift utilizzando gli agenti di estrazione dati AWS SCT
- Migrazione delle applicazioni legacy da Oracle Pro*C a ECPG
- Migra le colonne virtuali generate da Oracle a PostgreSQL
- Configura la funzionalità Oracle UTL_FILE su Aurora, compatibile con PostgreSQL
- Convalida gli oggetti del database dopo la migrazione da Oracle ad Amazon Aurora PostgreSQL

Converti VARCHAR2 (1) il tipo di dati per Oracle in tipo di dati booleano per Amazon Aurora PostgreSQL

Fotocamera Naresh, Amazon Web Services

Riepilogo

Durante una migrazione da Amazon Relational Database Service (Amazon RDS) per Oracle a Amazon Aurora PostgreSQL Compatible Edition, potresti riscontrare una mancata corrispondenza dei dati durante la convalida della migrazione in (). AWS Database Migration Service AWS DMS Per evitare questa mancata corrispondenza, puoi convertire il tipo di dati (1) in un tipo di dati booleano. VARCHAR2

VARCHAR2 il tipo di dati memorizza stringhe di testo a lunghezza variabile e VARCHAR2 (1) indica che la stringa è lunga 1 carattere o 1 byte. Per ulteriori informazioni su VARCHAR2, vedere i tipi di dati integrati in Oracle (documentazione Oracle).

In questo modello, nella colonna della tabella dei dati di origine di esempio, i dati VARCHAR2 (1) sono o Y, per Sì, o N, per No. Questo modello include istruzioni per utilizzare AWS DMS and AWS Schema Conversion Tool (AWS SCT) per convertire questo tipo di dati dai valori Y e N in VARCHAR2 (1) in valori veri o falsi in booleano.

Destinatari

Questo modello è consigliato a coloro che hanno esperienza nella migrazione di database Oracle ad Aurora compatibili con PostgreSQL utilizzando. AWS DMS Una volta completata la migrazione, segui i consigli in Conversione da Oracle ad Amazon RDS for PostgreSQL o Amazon Aurora PostgreSQL (documentazione).AWS SCT

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Verifica che il tuo ambiente sia pronto per Aurora, inclusa la configurazione di credenziali, autorizzazioni e un gruppo di sicurezza. Per ulteriori informazioni, consulta Configurazione dell'ambiente per Amazon Aurora (documentazione Aurora).
- Un database Amazon RDS for Oracle di origine che contiene una colonna di tabella VARCHAR2 con (1) dati.

- Un'istanza di database di destinazione compatibile con Amazon Aurora PostgreSQL. Per ulteriori informazioni, vedere Creazione di un cluster di database e connessione a un database su un cluster di database Aurora PostgreSQL (documentazione Aurora).

Versioni del prodotto

- Amazon RDS for Oracle versione 12.1.0.2 o successiva.
- AWS DMS versione 3.1.4 o successiva. Per ulteriori informazioni, vedere Utilizzo di un database Oracle come origine AWS DMS e Utilizzo di un database PostgreSQL come destinazione per (documentazione). AWS DMSAWS DMS Ti consigliamo di utilizzare la versione più recente di AWS DMS per il supporto più completo della versione e delle funzionalità.
- AWS Schema Conversion Tool (AWS SCT) versione 1.0.632 o successiva. Ti consigliamo di utilizzare la versione più recente di AWS SCT per il supporto più completo della versione e delle funzionalità.
- Aurora supporta le versioni di PostgreSQL elencate in Database Engine Versions for Aurora PostgreSQL compatible (documentazione Aurora).

Architecture

Stack tecnologico di origine

Istanza di database Amazon RDS per Oracle

Stack tecnologico Target

Istanza di database compatibile con Amazon Aurora PostgreSQL

Architettura di origine e destinazione

Tools (Strumenti)

Servizi AWS

- Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL.
- AWS Database Migration Service () ti aiuta a migrare gli archivi di dati all'interno o tra combinazioni di AWS DMS configurazioni cloud e locali. Cloud AWS

- Amazon Relational Database Service (Amazon RDS) per Oracle ti aiuta a configurare, gestire e scalare un database relazionale Oracle in. Cloud AWS
- AWS Schema Conversion Tool (AWS SCT) supporta migrazioni di database eterogenee convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione.

Altri servizi

- Oracle SQL Developer è un ambiente di sviluppo integrato che semplifica lo sviluppo e la gestione dei database Oracle nelle implementazioni tradizionali e basate sul cloud. In questo modello, utilizzi questo strumento per connetterti all'istanza di database Amazon RDS for Oracle e interrogare i dati.
- pgAdmin è uno strumento di gestione open source per PostgreSQL. Fornisce un'interfaccia grafica che consente di creare, gestire e utilizzare oggetti di database. In questo modello, si utilizza questo strumento per connettersi all'istanza del database Aurora e interrogare i dati.

Epiche

Preparati per la migrazione

Operazione	Description	Competenze richieste
Crea un rapporto sulla migrazione del database.	1. Nel AWS SCT, crea un rapporto di valutazione della migrazione del database. Per ulteriori informazioni, consulta Creazione di report di valutazione della migrazione. 2. Esamina ed esegui le azioni contenute nel rapporto di valutazione della migrazion e. Per ulteriori informazi oni, consulta le azioni del rapporto di valutazione.	DBA, Sviluppatore

Operazione	Description	Competenze richieste
Disabilita i vincoli di chiave esterna sul database di destinazione.	In PostgreSQL, le chiavi esterne vengono implement ate utilizzando i trigger. Durante la fase di caricamen to completo, AWS DMS carica ogni tabella una alla volta. Si consiglia vivamente di disabilitare i vincoli di chiave esterna durante un caricamen to completo utilizzando uno dei seguenti metodi: • Disabilitare temporane amente tutti i trigger dall'ista nza, quindi terminare il caricamento completo. • Utilizzare il parametro `session_replicatio n_role` in PostgreSQL. Se la disabilitazione dei vincoli di chiave esterna non è possibile, create un'attivi tà di AWS DMS migrazion e per i dati primari specifica della tabella principale e della tabella figlio.	DBA, Sviluppatore

Operazione	Description	Competenze richieste

Disabilita le chiavi primarie e le chiavi univoche sul database di destinazione.	Utilizzando i seguenti comandi, disabilitate le chiavi primarie e i vincoli sul database di destinazione. Ciò consente di migliorare le prestazioni dell'attività di caricamento iniziale.
	`ALTER TABLE <table> DISABLE PRIMARY KEY;`
	`ALTER TABLE <table> DISABLE CONSTRAINT <constraint_name>;`
		DBA, Sviluppatore

Crea l'attività di caricamento iniziale.	In AWS DMS, crea l'attività di migrazione per il caricamento iniziale. Per istruzioni, consulta Creazione di un'attività. Per il metodo di migrazione, scegli Migra dati esistenti. Questo metodo di migrazione viene richiamato `Full Load` nell'API. Non avviare ancora questa attività.	DBA, Sviluppatore

Operazione	Description	Competenze richieste
Modifica le impostazioni dell'attività per l'attività di caricamento iniziale.	Modifica le impostazioni dell'attività per aggiungere la convalida dei dati. Queste impostazioni di convalida devono essere create in un file JSON. Per istruzioni ed esempi, vedere Specifica zione delle impostazioni delle attività. Aggiungi le seguenti convalide: • Per verificare che i dati VARCHAR2 (1) vengano convertiti accuratamente in booleani nel database di destinazione, aggiungi il codice nello script di convalida dei dati nella sezione Informazioni aggiuntive di questo modello. Lo script di convalida converte i valori booleani da 1 a Y e da 0 a N nella tabella di destinazione, quindi confronta i valori nella tabella di destinazione con la tabella di origine. Per convalidare il resto della migrazione dei dati, abilita la convalida dei dati nell'atti vità. Per ulteriori informazioni, consulta Impostazioni dell'atti vità di convalida dei dati.	Amministratore AWS, DBA

Operazione	Description	Competenze richieste
Crea l'attività di replica continua.	In AWS DMS, crea l'attività di migrazione che mantiene il database di destinazione sincronizzato con il database di origine. Per istruzioni, consulta Creazione di un'attività. Per il metodo di migrazione, scegli Replica solo le modifiche ai dati. Non iniziate ancora questa attività.	DBA

Verifica le attività di migrazione

Operazione	Description	Competenze richieste
Crea dati di esempio per i test.	Nel database di origine, crea una tabella di esempio con dati a scopo di test.	Developer
Verifica che non vi siano attività in conflitto.	Utilizza il `pg_stat_activity` per verificare eventuali attività sul server che potrebbero influire sulla migrazione. Per ulteriori informazioni, vedere The Statistics Collector (documentazione PostgreSQL).	Amministratore AWS
Avvia le attività di AWS DMS migrazione.	Nella AWS DMS console, nella pagina Dashboard, avvia le attività di caricamento iniziale e di replica in corso che hai creato nell'epic precedente.	Amministratore AWS

Operazione	Description	Competenze richieste
Monitora le attività e gli stati di caricamento delle tabelle.	Durante la migrazione, monitora lo stato delle attività e gli stati della tabella. Una volta completata l'attività di caricamento iniziale, nella scheda Statistiche della tabella: • Lo stato di caricamento deve essere completato dalla tabella. • Lo stato di convalida deve essere convalidato.	Amministratore AWS
Verifica i risultati della migrazione.	Utilizzando pgAdmin, interroga la tabella sul database di destinazione. Una query riuscita indica che i dati sono stati migrati correttamente.	Developer
Aggiungi chiavi primarie e chiavi esterne nel database di destinazione.	Crea la chiave primaria e la chiave esterna nel database di destinazione. Per ulteriori informazioni, vedere ALTER TABLE (sito Web PostgreSQL).	DBA
Pulisci i dati del test.	Nei database di origine e di destinazione, pulisci i dati creati per il test unitario.	Developer

Tagliare

Operazione	Description	Competenze richieste
Completa la migrazione.	Ripeti l'epopea precedente, testa le attività di migrazion e utilizzando i dati di origine reale. In questo modo i dati vengono migrati dal database di origine a quello di destinazi one.	Developer
Verifica che i database di origine e di destinazione siano sincronizzati.	Verifica che i database di origine e di destinazione siano sincronizzati. Per ulteriori informazioni e istruzioni, consulta la sezione Convalida AWS DMS dei dati.	Developer
Arresta il database di origine.	Arresta il database Amazon RDS for Oracle. Per istruzion i, consulta Arresto temporane o di un'istanza DB Amazon RDS. Quando si arresta il database di origine, il caricamento iniziale e le attività di replica in corso AWS DMS vengono interrott e automaticamente. Non è richiesta alcuna azione aggiuntiva per interrompere queste attività.	Developer

Risorse correlate

AWS riferimenti

- Esegui la migrazione di un database Oracle ad Aurora AWS DMSAWS SCT PostgreSQL utilizzando e (Prescriptive Guidance)AWS
- Conversione da Oracle ad Amazon RDS per PostgreSQL o Amazon Aurora PostgreSQL (documentazione)AWS SCT
- AWS DMSAWS DMS Come funziona (documentazione)

Altri riferimenti

- Tipo di dati booleano (documentazione PostgreSQL)
- Tipi di dati Oracle integrati (documentazione Oracle)
- pgAdmin (sito web pgAdmin)
- SQL Developer (sito web Oracle)

Tutorial e video

- Guida introduttiva con AWS DMS
- Guida introduttiva ad Amazon RDS
- Introduzione a AWS DMS (video)
- Informazioni su Amazon RDS (video)

Informazioni aggiuntive

Script di convalida dei dati

Il seguente script di convalida dei dati converte 1 in Y e 0 in N. Ciò consente di completare correttamente l' AWS DMS attività e di superare la convalida della tabella.

```
{
"rule-type": "validation",
"rule-id": "5",
"rule-name": "5",
"rule-target": "column",
"object-locator": {
"schema-name": "ADMIN",
"table-name": "TEMP_CHRA_BOOL",
"column-name": "GRADE"
},
```

```
"rule-action": "override-validation-function",
"target-function": "case grade when '1' then 'Y' else 'N' end"
        }
```

L'`case`istruzione nello script esegue la convalida. Se la convalida fallisce, AWS DMS inserisce un record nella tabella public.awsdms_validation_failures_v1 sull'istanza del database di destinazione. Questo record include il nome della tabella, l'ora dell'errore e i dettagli sui valori non corrispondenti nelle tabelle di origine e di destinazione.

Se non si aggiunge questo script di convalida dei dati all' AWS DMS attività e i dati vengono inseriti nella tabella di destinazione, l' AWS DMS attività mostra lo stato di convalida come Record non corrispondenti.

Durante la AWS SCT conversione, il task di AWS DMS migrazione modifica il tipo di dati del tipo di dati VARCHAR2 (1) in Boolean e aggiunge un vincolo di chiave primaria sulla colonna. `''N0''`

Crea utenti e ruoli delle applicazioni in Aurora, compatibile con PostgreSQL

Abhishek Verma, Amazon Web Services

Riepilogo

Quando esegui la migrazione ad Amazon Aurora PostgreSQL Compatible Edition, gli utenti e i ruoli del database esistenti nel database di origine devono essere creati nel database compatibile con Aurora PostgreSQL. È possibile creare utenti e ruoli in Aurora compatibili con PostgreSQL utilizzando due approcci diversi:

- Utilizzate utenti e ruoli simili nel database di destinazione e nel database di origine. In questo approccio, i linguaggi di definizione dei dati (DDLs) vengono estratti per utenti e ruoli dal database di origine. Quindi vengono trasformati e applicati al database Aurora di destinazione compatibile con PostgreSQL. Ad esempio, il post sul blog Use SQL to map users, roles and grants from Oracle to PostgreSQL tratta l'utilizzo dell'estrazione da un motore di database sorgente Oracle.
- Utilizza utenti e ruoli standardizzati che vengono comunemente utilizzati durante lo sviluppo, l'amministrazione e per l'esecuzione di altre operazioni correlate nel database. Ciò include le operazioni di sola lettura, lettura/scrittura, di sviluppo, amministrazione e distribuzione eseguite dai rispettivi utenti.

Questo modello contiene le sovvenzioni necessarie per la creazione di utenti e ruoli in Aurora, compatibile con PostgreSQL, necessarie per l'approccio standardizzato di utenti e ruoli. Le fasi di creazione di utenti e ruoli sono allineate alla politica di sicurezza che prevede la concessione dei privilegi minimi agli utenti del database. La tabella seguente elenca gli utenti, i ruoli corrispondenti e i relativi dettagli nel database.

Utenti	Roles	Scopo
`APP_read`	`APP_RO`	Utilizzato per l'accesso in sola lettura allo schema `APP`
`APP_WRITE`	`APP_RW`	Utilizzato per le operazioni di scrittura e lettura sullo schema `APP`
`APP_dev_user`	`APP_DEV`	Utilizzato a scopo di sviluppo sullo schema`APP_DEV`, con

		accesso in sola lettura allo schema APP
Admin_User	rds_superuser	Utilizzato per eseguire operazioni di amministratore sul database
APP	APP_DEP	Utilizzato per creare gli oggetti nell'ambito dello APP schema e per la distribuzione degli oggetti nello APP schema

Prerequisiti e limitazioni

Prerequisiti

- Un account Amazon Web Services (AWS) attivo
- Un database PostgreSQL, un database Edition compatibile con Amazon Aurora PostgreSQL o un database Amazon Relational Database Service (Amazon RDS) per PostgreSQL

Versioni del prodotto

- Tutte le versioni di PostgreSQL

Architecture

Stack tecnologico di origine

- Qualsiasi database

Stack tecnologico Target

- Compatibile con Amazon Aurora PostgreSQL

Architettura Target

Il diagramma seguente mostra i ruoli utente e l'architettura dello schema nel database Aurora compatibile con PostgreSQL.

Automazione e scalabilità

Questo modello contiene gli utenti, i ruoli e lo script di creazione dello schema, che è possibile eseguire più volte senza alcun impatto sugli utenti esistenti del database di origine o di destinazione.

Tools (Strumenti)

Servizi AWS

- Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL.

Altri servizi

- psql è uno strumento front-end basato su terminale che viene installato con ogni installazione del database PostgreSQL. Dispone di un'interfaccia a riga di comando per l'esecuzione di comandi SQL, PL-PGSQL e del sistema operativo.
- pgAdmin è uno strumento di gestione open source per PostgreSQL. Fornisce un'interfaccia grafica che consente di creare, gestire e utilizzare oggetti di database.

Epiche

Crea gli utenti e i ruoli

Operazione	Description	Competenze richieste
Crea l'utente di distribuzione.	L'utente di distribuzione APP verrà utilizzato per creare e modificare gli oggetti del database durante le distribuz ioni. Utilizza gli script seguenti per creare il ruolo utente di distribuzione APP_DEP nello schema. APP Convalida i diritti	DBA

Operazione	Description	Competenze richieste

	di accesso per assicurarti che questo utente abbia solo il privilegio di creare oggetti nello schema richiesto. APP
	
	1. Connect all'utente admin e crea lo schema.
	
	`CREATE SCHEMA APP;`
	
	2. Creare l'utente.
	
	`CREATE USER APP WITH PASSWORD <password> ;`
	
	3. Crea il ruolo.
	
	`CREATE ROLE APP_DEP ;`
	`GRANT all on schema APP to APP_DEP ;`
	`GRANT USAGE ON SCHEMA APP to APP_DEP ;`
	`GRANT connect on database <db_name> to APP_DEP ;`
	`GRANT APP_DEP to APP;`
	
	4. Per testare i privilegi, connettiti APP e crea le tabelle.
	
	`set search_path to APP;`
	`SET`
	`CREATE TABLE test(id integer ) ;`
	`CREATE TABLE`
	
	5. Controlla i privilegi.	

Operazione	Description	Competenze richieste
	`select schemaname , tablename , tableowner from pg_tables where tablename like 'test' ; schemaname | tablename | tableowner APP | test | APP`	

Operazione	Description	Competenze richieste

Crea l'utente di sola lettura.	L'utente di sola lettura `APP_read` verrà utilizzato per eseguire l'operazione di sola lettura nello schema. APP Utilizzare gli script seguenti per creare l'utente di sola lettura. Convalida i diritti di accesso per assicurarti che questo utente abbia il privilegio solo di leggere gli oggetti nello schema APP e di concedere automaticamente l'accesso in lettura per tutti i nuovi oggetti creati nello schema. APP
	
	1. Crea l'utente. `APP_read`
	
	`create user APP_read ; alter user APP_read with password 'your_password' ;`
	
	2. Crea il ruolo.
	
	`CREATE ROLE APP_ro ; GRANT SELECT ON ALL TABLES IN SCHEMA APP TO APP_RO ; GRANT USAGE ON SCHEMA APP TO APP_RO GRANT CONNECT ON DATABASE testdb TO APP_RO ; GRANT APP_RO TO APP_read;`
		DBA

Operazione	Description	Competenze richieste

	3. Per testare i privilegi , accedi utilizzando l'APP_readutente.
	
	`set search_path to APP ;`
	`create table test1( id integer) ;`
	`ERROR: permission denied for schema APP`
	`LINE 1: create table test1( id integer) ;`
	`insert into test values (34) ;`
	`ERROR: permission denied for table test`
	`SQL state: 42501`
	`select from test`
	`no rows selected`	

Operazione	Description	Competenze richieste

Crea l' read/write utente.	L' read/write utente `APP_WRITE` verrà utilizzato per eseguire operazioni di lettura e scrittura sullo schemaAPP. Usa i seguenti script per creare l' read/write utente e concedergli il `APP_RW` ruolo. Convalida i diritti di accesso per assicurarti che questo utente disponga dei privilegi di lettura e scrittura solo sugli oggetti dello schema `APP` e per concedere automaticamente l'accesso in lettura e scrittura per ogni nuovo oggetto creato nello schema. `APP`
	
	1. Creare l’utente.
	
	`CREATE USER APP_WRITE ; alter user APP_WRITE with password 'your_password' ;`
	
	2. Crea il ruolo.
	
	`CREATE ROLE APP_RW; GRANT SELECT, INSERT, UPDATE, DELETE ON ALL TABLES IN SCHEMA APP TO APP_RW ; GRANT CONNECT ON DATABASE postgres to APP_RW ; GRANT USAGE ON SCHEMA APP to APP_RW ;`	

Operazione	Description	Competenze richieste
	ALTER DEFAULT PRIVILEGES IN SCHEMA APP GRANT SELECT, INSERT, UPDATE, DELETE ON TABLES TO APP_RW ; GRANT APP_RW to APP_WRITE 3. Per testare i privilegi, accedi utilizzando l'utente. APP_WRITE SET SEARCH_PATH to APP; CREATE TABLE test1(id integer) ; ERROR: permission denied for schema APP LINE 1: create table test1(id integer) ; SELECT * FROM test ; id ---- 12 INSERT INTO test values (31) ; INSERT 0 1	

Operazione	Description	Competenze richieste

Crea l'utente amministratore.	L'utente amministratore `Admin_User` verrà utilizzato per eseguire operazioni di amministrazione sul database. Esempi di queste operazioni sono `CREATE ROLE` e`CREATE DATABASE`. `Admin_User` utilizza il ruolo integrato `rds_superuser` per eseguire operazioni di amministrazione sul database. Utilizza gli script seguenti per creare e testare il privilegio per l'utente amministratore `Admin_User` nel database.
	1. Crea l'utente e concedi il ruolo.
	`create user Admin_User WITH PASSWORD ‘Your password’ ALTER user Admin_user CREATEDB; ALTER user Admin_user CREATEROLE;`
	2. Per testare il privilegio, accedi dall'`Admin_User` utente.
	`SELECT * FROM APP.test ; id ---- 31 CREATE ROLE TEST ;`
		DBA

Operazione	Description	Competenze richieste
	`CREATE DATABASE test123 ;`	

Operazione	Description	Competenze richieste

Crea l'utente di sviluppo.	L'utente di sviluppo `APP_dev_user` avrà il diritto di creare gli oggetti nel suo schema locale `APP_DEV` e di leggere l'accesso allo schemaAPP. Utilizzate i seguenti script per creare e testare i privilegi dell'utente `APP_dev_user` nel database.
	
	1. Creare l'utente.
	
	`CREATE USER APP1_dev_user with password 'your password';`
	
	2. Crea lo `APP_DEV` schema per. `App_dev_user`
	
	`CREATE SCHEMA APP1_DEV ;`
	
	3. Crea il APP_DEV ruolo.
	
	`CREATE ROLE APP1_DEV ;`
	`GRANT APP1_RO to APP1_DEV ;`
	`GRANT SELECT ON ALL TABLES IN SCHEMA APP1_DEV to APP1_dev_user`
	`GRANT USAGE, CREATE ON SCHEMA APP1_DEV to APP1_DEV_USER`
	`GRANT APP1_DEV to APP1_DEV_USER ;`
		DBA

Operazione	Description	Competenze richieste

	4. Per testare i privilegi, accedi daAPP_dev_user .
	
	`CREATE TABLE APP1_dev.test1( id integer ) ;`
	`CREATE TABLE`
	`INSERT into APP1_dev.test1 ( select * from APP1.test );`
	`INSERT 0 1`
	`CREATE TABLE APP1.test4 ( id int) ;`
	`ERROR: permissio n denied for schema APP1`
	`LINE 1: create table APP1.test4 ( id int) ;`	

Risorse correlate

Documentazione PostgreSQL

- CREA RUOLO
- CREA UTENTE
- Ruoli predefiniti

Informazioni aggiuntive

Miglioramento di PostgreSQL 14

PostgreSQL 14 fornisce una serie di ruoli predefiniti che danno accesso a determinate funzionalità e informazioni privilegiate di uso comune. Gli amministratori (compresi i ruoli che dispongono del `CREATE ROLE` privilegio) possono concedere questi ruoli o altri ruoli nel loro ambiente agli utenti, fornendo loro l'accesso alle funzionalità e alle informazioni specificate.

Gli amministratori possono concedere agli utenti l'accesso a questi ruoli utilizzando il comando. `GRANT` Ad esempio, per concedere il `pg_signal_backend` ruolo a`Admin_User`, è possibile eseguire il comando seguente.

```
GRANT pg_signal_backend TO Admin_User;
```

Il `pg_signal_backend` ruolo ha lo scopo di consentire agli amministratori di abilitare ruoli affidabili e non superutente per inviare segnali ad altri backend. Per ulteriori informazioni, consulta Miglioramento di PostgreSQL 14.

Ottimizzazione dell'accesso

In alcuni casi, potrebbe essere necessario fornire un accesso più granulare agli utenti (ad esempio, accesso basato su tabelle o accesso basato su colonne). In questi casi, è possibile creare ruoli aggiuntivi per concedere tali privilegi agli utenti. Per ulteriori informazioni, vedere PostgreSQL Grants.

Emula Oracle DR utilizzando un database globale Aurora compatibile con PostgreSQL

HariKrishna Boorgadda, Amazon Web Services

Riepilogo

Le migliori pratiche per il disaster recovery (DR) aziendale consistono fondamentalmente nella progettazione e implementazione di sistemi hardware e software con tolleranza ai guasti in grado di sopravvivere a un disastro (continuità aziendale) e riprendere le normali operazioni (ripresa aziendale), con un intervento minimo e, idealmente, senza perdita di dati. La creazione di ambienti con tolleranza ai guasti per soddisfare gli obiettivi aziendali di disaster recovery può essere costosa e dispendiosa in termini di tempo e richiede un forte impegno da parte dell'azienda.

Oracle Database offre tre diversi approcci al DR che offrono il massimo livello di protezione e disponibilità dei dati rispetto a qualsiasi altro approccio per la protezione dei dati Oracle.

- Dispositivo di ripristino Oracle Zero Data Loss
- Oracle Active Data Guard
- Oracle GoldenGate

Questo modello fornisce un modo per emulare Oracle GoldenGate DR utilizzando un database globale Amazon Aurora. L'architettura di riferimento utilizza Oracle GoldenGate for DR in tre regioni AWS. Lo schema illustra la ripiattaforma dell'architettura di origine nel database globale Aurora nativo per il cloud basato su Amazon Aurora PostgreSQL — Compatible Edition.

I database globali Aurora sono progettati per applicazioni con un'impronta globale. Un singolo database Aurora si estende su più regioni AWS con un massimo di cinque regioni secondarie. I database globali Aurora offrono le seguenti funzionalità:

- Replica fisica a livello di storage
- Letture globali a bassa latenza
- Ripristino di emergenza rapido in caso di interruzioni a livello regionale
- Migrazioni rapide tra regioni
- Basso ritardo di replica tra le regioni
- Little-to-no impatto sulle prestazioni del database

Per ulteriori informazioni sulle caratteristiche e i vantaggi dei database globali di Aurora, consulta Utilizzo dei database globali di Amazon Aurora. Per ulteriori informazioni sui failover non pianificati e gestiti, consulta Uso del failover in un database globale Amazon Aurora.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un driver PostgreSQL Java Database Connectivity (JDBC) per la connettività delle applicazioni
- Un database globale Aurora basato su Amazon Aurora PostgreSQL Compatible Edition
- Un database Oracle Real Application Clusters (RAC) migrato al database globale Aurora basato sulla compatibilità con Aurora PostgreSQL

Limitazioni dei database globali Aurora

- I database globali Aurora non sono disponibili in tutte le regioni AWS. Per un elenco delle regioni supportate, consulta Database globali Aurora con Aurora PostgreSQL.
- Per informazioni sulle funzionalità non supportate e altre limitazioni dei database globali di Aurora, consulta le Limitazioni dei database globali di Amazon Aurora.

Versioni del prodotto

- Amazon Aurora PostgreSQL — Compatible Edition versione 10.14 o successiva

Architecture

Stack tecnologico di origine

- Database Oracle RAC a quattro nodi
- Oracle GoldenGate

Architettura di origine

Il diagramma seguente mostra tre cluster con Oracle RAC a quattro nodi in diverse regioni AWS replicati utilizzando Oracle. GoldenGate

Stack tecnologico Target

- Un database globale Amazon Aurora a tre cluster basato su Aurora PostgreSQL, compatibile con un cluster nella regione principale, due cluster in diverse regioni secondarie

Architettura Target

Tools (Strumenti)

Servizi AWS

- Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL.
- I database globali di Amazon Aurora si estendono su più regioni AWS, fornendo letture globali a bassa latenza e ripristino rapido da rare interruzioni che potrebbero interessare un'intera regione AWS.

Epiche

Aggiungi regioni con istanze DB Reader

Operazione	Description	Competenze richieste
Collega uno o più cluster Aurora secondari.	Nella Console di gestione AWS, scegli Amazon Aurora. Seleziona il cluster primario, scegli Azioni e scegli Aggiungi regione dall'elenco a discesa.	DBA
Seleziona la classe dell'istanza.	È possibile modificare la classe di istanza del cluster secondario. Tuttavia, si consiglia di mantenerla uguale alla classe di istanza del cluster principale.	DBA

Operazione	Description	Competenze richieste
Aggiungi la terza regione.	Ripeti i passaggi di questa epopea per aggiungere un cluster nella terza regione.	DBA

Esegui il failover del database globale Aurora

Operazione	Description	Competenze richieste
Rimuovi il cluster primario dal database globale Aurora.	1. Nella pagina Database, scegli il cluster primario. 2. Scegli Rimuovi da globale per eseguire il failover su un cluster secondario.	DBA
Riconfigurare l'applicazione per deviare il traffico di scrittura nel nuovo cluster promosso.	Modifica l'endpoint nell'appl icazione con quello del cluster appena promosso.	DBA
Interrompi l'esecuzione di operazioni di scrittura sul cluster non disponibile.	Interrompi l'applicazione e qualsiasi attività DML (Data Manipulation Language) sul cluster che hai rimosso.	DBA
Crea un nuovo database globale Aurora.	Ora puoi creare un database globale Aurora con il cluster appena promosso come cluster primario.	DBA

Avvia il cluster primario

Operazione	Description	Competenze richieste
Seleziona il cluster primario da avviare dal database globale.	Nella console Amazon Aurora, nella configurazione di Global Database, scegli il cluster primario.	DBA
Avvia il cluster.	Nell'elenco a discesa Azioni, scegli Avvia. Questo processo potrebbe richiedere del tempo. Aggiorna la schermata per visualizzare lo stato oppure controlla la colonna Status per lo stato corrente del cluster una volta completata l'operazione.	DBA

Pulisci le risorse

Operazione	Description	Competenze richieste
Eliminare i cluster secondari rimanenti.	Una volta completato il programma pilota di failover, rimuovete i cluster secondari dal database globale.	DBA
Eliminare il cluster primario.	Rimuovi il cluster.	DBA

Risorse correlate

- Using Amazon Aurora global databases
- Soluzioni di disaster recovery Aurora PostgreSQL che utilizzano Amazon Aurora Global Database (post sul blog)

SHA1 Implementa l'hashing per i dati PII durante la migrazione da SQL Server a PostgreSQL

Rajkumar Raghuwanshi e Jagadish Kantubugata, Amazon Web Services

Riepilogo

Questo modello descrive come implementare l'hashing Secure Hash Algorithm 1 (SHA1) per gli indirizzi e-mail durante la migrazione da SQL Server ad Amazon RDS for PostgreSQL o Amazon Aurora PostgreSQL compatibile. Un indirizzo e-mail è un esempio di informazioni di identificazione personale (PII). Le PII sono informazioni che, se visualizzate direttamente o abbinate ad altri dati correlati, possono essere utilizzate per dedurre ragionevolmente l'identità di un individuo.

Questo modello copre le sfide legate al mantenimento di valori hash coerenti tra diverse regole di confronto di database e codifiche di caratteri e fornisce una soluzione che utilizza funzioni e trigger PostgreSQL. Sebbene questo modello si concentri sull' SHA1 hashing, può essere adattato per altri algoritmi di hashing supportati dal modulo di PostgreSQL. `pgcrypto` Considerate sempre le implicazioni di sicurezza della vostra strategia di hashing e consultate gli esperti di sicurezza se gestite dati sensibili.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Database SQL Server di origine
- Database PostgreSQL di destinazione (compatibile con Amazon RDS per PostgreSQL o Aurora PostgreSQL)
- Esperienza nella codifica PL/PgSQL

Limitazioni

- Questo modello richiede modifiche alle regole di confronto a livello di database in base ai casi d'uso.
- L'impatto sulle prestazioni su set di dati di grandi dimensioni non è stato valutato.
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità regionale, vedi AWS Servizi per regione. Per endpoint specifici, consulta Service endpoints and quotas e scegli il link relativo al servizio.

Versioni del prodotto

- Microsoft SQL Server 2012 o versioni successive

Architecture

Stack tecnologico di origine

- SQL Server
- .NET Framework

Stack tecnologico di destinazione

- PostgreSQL
- `pgcrypto`estensione

Automazione e scalabilità

- Prendi in considerazione l'implementazione della funzione di hashing come procedura memorizzata per semplificare la manutenzione.
- Per set di dati di grandi dimensioni, valuta le prestazioni e prendi in considerazione l'elaborazione in batch o le strategie di indicizzazione.

Tools (Strumenti)

Servizi AWS

- Amazon Aurora PostgreSQL Compatible è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL.
- AWS Database Migration Service () ti aiuta a migrare gli archivi di dati all'interno o tra combinazioni di AWS DMS configurazioni cloud e locali. Cloud AWS
- Amazon Relational Database Service Amazon RDS per PostgreSQL ti aiuta a configurare, gestire e scalare un database relazionale PostgreSQL in. Cloud AWS
- AWS Schema Conversion Tool (AWS SCT) supporta migrazioni di database eterogenee convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione.

Altri strumenti

- pgAdmin è uno strumento di gestione open source per PostgreSQL. Fornisce un'interfaccia grafica che consente di creare, gestire e utilizzare oggetti di database.
- SQL Server Management Studio (SSMS) è un ambiente integrato per la gestione di qualsiasi infrastruttura SQL.

Best practice

- Utilizza le impostazioni di confronto appropriate per gestire i caratteri speciali sul lato del database di destinazione.
- Esegui test approfonditi con una varietà di indirizzi e-mail, inclusi gli indirizzi con caratteri non ASCII.
- Mantieni la coerenza nella gestione delle lettere maiuscole e minuscole tra i livelli dell'applicazione e del database.
- Esegui il benchmark delle prestazioni delle query utilizzando i valori con hash.

Epiche

Analizza l'implementazione dell'hashing dei sorgenti

Operazione	Description	Competenze richieste
Rivedi il codice di SQL Server.	Per esaminare il codice di SQL Server che genera gli SHA1 hash, procedi come segue: • Analizza l'implementazione esistente dell' SHA1 hashing in SQL Server. • Identifica i metodi esatti utilizzati per la generazione di hash. • Documenta i parametri di input e il formato di output.	Ingegnere dei dati, DBA, sviluppatore di app

Operazione	Description	Competenze richieste
	• Esamina eventuali conversioni o trasformazioni dei tipi di dati. • Esamina le impostazioni di confronto e il loro impatto.	
Documenta l'algoritmo di hashing e le trasformazioni dei dati.	Per documentare l'esatto algoritmo di hashing e le trasformazioni dei dati, procedi come segue: • Crea una documentazione tecnica dettagliata del processo di hashing. • Documenta la logica di step-by-step trasformazione. • Specificate i formati di input e output e i tipi di dati. • Include i casi limite e la gestione dei caratteri speciali.	Sviluppatore di app, ingegnere dei dati, DBA

Crea una funzione di hashing PostgreSQL

Operazione	Description	Competenze richieste
Crea `pgcrypto` estensione.	Per creare l'`pgcrypto`estensione, usa `pgAdmin/psql` per eseguire il seguente comando: `CREATE EXTENSION pgcrypto;`	DBA, ingegnere dei dati

Operazione	Description	Competenze richieste
Implementa una funzione PostgreSQL.	Implementa la seguente funzione PostgreSQL per replicare la logica di hashing di SQL Server. A un livello elevato, questa funzione utilizza i seguenti passaggi: 1. Facoltativamente, converte l'input in lettere maiuscole. 2. Crea un SHA1 hash dell'input. 3. Richiede gli ultimi 10 byte (80 bit) di questo hash. 4. Converte questi byte in un numero intero a 64 bit. `CREATE OR REPLACE FUNCTION utility.hex_to_bigint ( par_val character varying, par_upper character varying DEFAULT 'lower'::character varying) RETURNS bigint LANGUAGE 'plpgsql' AS $BODY$ DECLARE retnumber bigint; digest_bytes bytea; BEGIN if lower(par_upper) = 'upper' then`	Ingegnere dei dati, DBA, sviluppatore di app

Operazione	Description	Competenze richieste
	`digest_bytes := digest(upper(par_val), 'sha1'); else digest_bytes := digest((par_val), 'sha1'); end if; retnumber := ('x' || encode(substring(digest_bytes, length(digest_bytes)-10+1), 'hex'))::bit(64)::bigint; RETURN retnumber; END; $BODY$;`	

Operazione	Description	Competenze richieste

Esegui il test della funzione.	Per testare la funzione, utilizza dati di esempio provenienti da SQL Server per verificare i valori hash corrispondenti. Esegui il comando seguente:
	`select 'alejandr o_rosalez@example. com' as Email, utility.hex_to_big int('alejandro_ros alez@example.com', 'upper') as HashValue;`
	`--OUTPUT /* email hashvalue "alejandro_rosale z@example.com" 451 397011176045063 */`
		Sviluppatore di app, DBA, ingegnere dei dati

Implementa i trigger per l'hashing automatico

Operazione	Description	Competenze richieste
Crea trigger su tabelle pertinenti.	Per creare trigger nelle tabelle pertinenti per generare automaticamente valori hash all'inserimento o all'aggio rnamento, esegui il comando seguente: `CREATE OR REPLACE FUNCTION update_em ail_hash()`	Sviluppatore di app, ingegnere dei dati, DBA

Operazione	Description	Competenze richieste

	`RETURNS TRIGGER AS $$ BEGIN NEW.email_hash = utility.hex_to_bigint(NEW.email, 'upper'); RETURN NEW; END; $$ LANGUAGE plpgsql;`
	
	`CREATE TRIGGER email_hash_trigger BEFORE INSERT OR UPDATE ON users FOR EACH ROW EXECUTE FUNCTION update_email_hash();`	

Migrazione dei dati esistenti

Operazione	Description	Competenze richieste
Sviluppa o utilizza uno script di migrazione AWS DMS.	Sviluppa uno script di migrazione o AWS DMS utilizzalo per compilare i valori hash per i dati esistenti (inclusi i valori hash memorizzati come `BIGINT` nel sistema di origine). Completa le seguenti operazioni: • Crea script di migrazione per il trasferimento dei dati con valori hash.	Ingegnere dei dati, sviluppatore di app, DBA

Operazione	Description	Competenze richieste
	• Configura AWS DMS le attività con regole di trasformazione appropriate. • Configura gli endpoint di origine e di destinazione in AWS DMS. • Implementa meccanismi di gestione e registrazione degli errori. • Progetta una strategia di elaborazione in batch per set di dati di grandi dimensioni. • Crea query di convalida per la verifica dei dati.	
Usa la nuova funzione di hashing PostgreSQL.	Per utilizzare la nuova funzione di hashing PostgreSQL per garantire la coerenza, procedi come segue: • Implementa procedure di convalida per verificare la coerenza dell'hash. • Crea script di confronto tra i sistemi di origine e di destinazione. • Imposta test automatici per la verifica del valore hash. • Documenta eventuali discrepanze e le fasi di risoluzione.	Sviluppatore di app, DBA, ingegnere DevOps

Aggiorna le interrogazioni delle applicazioni

Operazione	Description	Competenze richieste
Identifica le domande relative all'applicazione.	Per identificare le query dell'applicazione che utilizzano valori con hash, effettuate le seguenti operazioni: • Analizzate la base di codice dell'applicazione per le query che utilizzano valori hash. • Esamina le procedure e le funzioni memorizzate che fanno riferimento alle operazioni di hash. • Documenta le metriche delle prestazioni delle query e i piani di esecuzione. • Identifica le dipendenze dalle ricerche basate su hash. • Mappa i componenti dell'applicazione interessati.	Sviluppatore di app, DBA, ingegnere dei dati
Modifica le interrogazioni.	Se necessario, modifica le query per utilizzare la nuova funzione di hashing PostgreSQL. Esegui questa operazione: • Rifattorizza le query esistenti per utilizzare la funzione di hashing PostgreSQL.	Sviluppatore di app, DBA, ingegnere dei dati

Operazione	Description	Competenze richieste
	• Aggiorna le procedure e le funzioni memorizzate. • Implementa e testa nuovi modelli di interrogazione. • Ottimizza le query modificate per migliorarne le prestazioni.	

Testa e convalida

Operazione	Description	Competenze richieste
Eseguire test.	Per eseguire test approfonditi con un sottoinsieme di dati di produzione, procedi come segue: • Crea un piano di test per la convalida dei sottoinsiemi di dati. • Estrai un campione rappresentativo di dati di produzione. • Configura un ambiente di test con configurazioni appropriate. • Esegui test di caricamento e trasformazione dei dati. • Esegui test di volume e stress.	Sviluppatore di app, ingegnere dei dati, DBA
Verifica che i valori hash corrispondano.	Per verificare che i valori hash corrispondano tra SQL Server	Sviluppatore di app, ingegnere dei dati, DBA

Operazione	Description	Competenze richieste
	e PostgreSQL, procedi come segue: • Sviluppa script di confronto per i valori hash. • Crea report di convalida per la corrispondenza degli hash. • Implementa procedure di verifica automatizzate. • Documenta eventuali discrepanze rilevate. • Analizza e risolvi le discrepanze di hash.	
Verifica la funzionalità dell'applicazione.	Per verificare la funzionalità dell'applicazione utilizzando i dati migrati e la nuova implementazione di hashing, procedi come segue: • Eseguite il test end-to-end delle applicazioni. • Convalida tutte le funzionalità dell'applicazione utilizzando dati crittografati. • Testa le prestazioni delle applicazioni con la nuova implementazione. • Verifica le integrazioni e le dipendenze delle API.	Sviluppatore di app, DBA, ingegnere dei dati

risoluzione dei problemi

Problema	Soluzione
I valori hash non corrispondono.	Verifica le codifiche e le regole di confronto dei caratteri tra origine e destinazione. Per ulteriori informazioni, consulta Gestire le modifiche alle regole di confronto in PostgreSQL su Amazon Aurora e Amazon RDS (Blog).AWS

Risorse correlate

AWS Blog

- Gestisci le modifiche alle regole di confronto in PostgreSQL su Amazon Aurora e Amazon RDS
- Esegui la migrazione di SQL Server ad Amazon Aurora PostgreSQL utilizzando le migliori pratiche e le lezioni apprese sul campo

Altre risorse

- Modulo PostgreSQL pgcrypto (documentazione PostgreSQL)
- Funzioni di attivazione di PostgreSQL (documentazione PostgreSQL)
- Funzione HASHBYTES di SQL Server (documentazione Microsoft)

Migrazione incrementale da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL utilizzando Oracle SQL Developer e AWS SCT

Pinesh Singal, Amazon Web Services

Riepilogo

Molte strategie e approcci di migrazione si svolgono in più fasi che possono durare da alcune settimane a diversi mesi. Durante questo periodo, è possibile che si verifichino ritardi dovuti all'applicazione di patch o aggiornamenti nelle istanze database Oracle di origine che si desidera migrare su istanze DB PostgreSQL. Per evitare questa situazione, si consiglia di migrare in modo incrementale il codice del database Oracle rimanente al codice del database PostgreSQL.

Questo modello fornisce una strategia di migrazione incrementale senza tempi di inattività per un'istanza DB Oracle da più terabyte che ha un numero elevato di transazioni eseguite dopo la migrazione iniziale e che deve essere migrata su un database PostgreSQL. Puoi utilizzare l' step-by-stepapproccio di questo modello per migrare in modo incrementale un'istanza DB Amazon Relational Database Service (Amazon RDS) per Oracle DB verso un'istanza DB Amazon RDS for PostgreSQL senza accedere alla console di gestione Amazon Web Services (AWS).

Il modello utilizza Oracle SQL Developer per trovare le differenze tra due schemi nel database Oracle di origine. Utilizza quindi AWS Schema Conversion Tool (AWS SCT) per convertire gli oggetti dello schema del database Amazon RDS for Oracle in oggetti dello schema del database Amazon RDS for PostgreSQL. È quindi possibile eseguire uno script Python nel prompt dei comandi di Windows per creare oggetti AWS SCT per le modifiche incrementali agli oggetti del database di origine.

> **Note**
>
> Prima di migrare i carichi di lavoro di produzione, ti consigliamo di eseguire un proof of concept (PoC) per l'approccio di questo modello in un ambiente di test o non di produzione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un'istanza database Amazon RDS for Oracle esistente.
- Un'istanza database Amazon RDS for PostgreSQL esistente.

- AWS SCT, installato e configurato con driver JDBC per motori di database Oracle e PostgreSQL. Per ulteriori informazioni su questo argomento, consulta Installazione di AWS SCT e Installazione dei driver di database richiesti nella documentazione di AWS SCT.
- Oracle SQL Developer, installato e configurato. Per ulteriori informazioni su questo argomento, vedere la documentazione di Oracle SQL Developer.
- Il `incremental-migration-sct-sql.zip` file (allegato), scaricato sul computer locale.

Limitazioni

- I requisiti minimi per l'istanza DB di origine di Amazon RDS for Oracle sono:
  - Versioni Oracle 10.2 e successive (per le versioni 10.x), 11g (versioni 11.2.0.3.v1 e successive) e fino a 12.2 e 18c per le edizioni Enterprise, Standard, Standard One e Standard Two
- I requisiti minimi per l'istanza database Amazon RDS for PostgreSQL di destinazione sono:
  - PostgreSQL versioni 9.4 e successive (per le versioni 9.x), 10.x e 11.x
- Questo modello utilizza Oracle SQL Developer. I risultati potrebbero variare se si utilizzano altri strumenti per trovare ed esportare le differenze dello schema.
- Gli script SQL generati da Oracle SQL Developer possono generare errori di trasformazione, il che significa che è necessario eseguire una migrazione manuale.
- Se le connessioni di test di origine e destinazione di AWS SCT falliscono, assicurati di configurare le versioni dei driver JDBC e le regole in entrata per il gruppo di sicurezza del cloud privato virtuale (VPC) per accettare il traffico in entrata.

Versioni del prodotto

- Istanza Amazon RDS for Oracle DB versione 12.1.0.2 (versione 10.2 e successive)
- Istanza database Amazon RDS per PostgreSQL versione 11.5 (versione 9.4 e successive)
- Oracle SQL Developer versione 19.1 e successive
- AWS SCT versione 1.0.632 e successive

Architecture

Stack tecnologico di origine

- Istanza database Amazon RDS per Oracle

Stack tecnologico Target

- Istanza database Amazon RDS per PostgreSQL

Architettura di origine e destinazione

Il diagramma seguente mostra la migrazione di un'istanza DB Amazon RDS for Oracle verso un'istanza DB Amazon RDS for PostgreSQL.

Il diagramma mostra il seguente flusso di lavoro di migrazione:

1. Aprire Oracle SQL Developer e connettersi ai database di origine e di destinazione.
2. Genera un report sulle differenze e quindi genera il file di script SQL per gli oggetti di differenza dello schema. Per ulteriori informazioni sui report sulle differenze, consulta Rapporti sulle differenze dettagliati nella documentazione di Oracle.
3. Configura AWS SCT ed esegui il codice Python.
4. Il file di script SQL converte da Oracle a PostgreSQL.
5. Esegui il file di script SQL sull'istanza database PostgreSQL di destinazione.

Automazione e scalabilità

Puoi automatizzare questa migrazione aggiungendo parametri aggiuntivi e modifiche relative alla sicurezza per più funzionalità in un unico programma allo script Python.

Tools (Strumenti)

- AWS SCT — AWS Schema Conversion Tool (AWS SCT) converte lo schema di database esistente da un motore di database a un altro.
- Oracle SQL Developer — Oracle SQL Developer è un ambiente di sviluppo integrato (IDE) che semplifica lo sviluppo e la gestione dei database Oracle nelle implementazioni tradizionali e basate sul cloud.

Codice

Il `incremental-migration-sct-sql.zip` file (allegato) contiene il codice sorgente completo per questo modello.

Epiche

Crea il file di script SQL per le differenze dello schema del database di origine

Operazione	Description	Competenze richieste
Esegui Database Diff in Oracle SQL Developer.	1. Accedi all'istanza Oracle DB di origine, scegli Strumenti, quindi scegli Database Diff. 2. Scegli il tuo database di origine in Source Connectio n. 3. Scegli il database di origine aggiornato o con patch in Destination Connection. 4. Configura le opzioni rimanenti in base alle tue esigenze, scegli Avanti, quindi scegli Fine per generare il rapporto sulle differenze.	DBA
Genera il file di script SQL.	Scegliete Genera script per generare le differenze nei file SQL. Questo genera il file di script SQL che AWS SCT utilizza per convertire il database da Oracle a PostgreSQL.	DBA

Usa lo script Python per creare gli oggetti DB di destinazione in AWS SCT

Operazione	Description	Competenze richieste
Configura AWS SCT con il prompt dei comandi di Windows.	1. Copia il `AWSSchemaConversionToolBatch.jar` file dalla cartella AWS SCT preinstallata e incollalo nella tua directory di lavoro. 2. Distribuisci il codice Python dal file `run_aws_sct_sql.py` dalla cartella (`incremental-migration-sct-sql.zip` allegato). Questo crea file.xml e.sct nella `projects` directory con i dettagli di configurazione dell'ambiente del database di origine e di destinazione. Legge anche il file di script SQL generato in Oracle SQL Developer. Infine, crea oggetti di file.sql nella directory. `output` 3. Configura i dettagli di configurazione dell'ambiente di origine e di destinazione nel `database_migration.txt` file utilizzando il seguente formato: `#source_vendor,source_hostname,sourc`	DBA

Operazione	Description	Competenze richieste
	e_dbname,source_us er,source_pwd,sour ce_schema,source_p ort,source_sid,tar get_vendor,target_ hostname,target_us er,target_pwd,targ et_dbname,target_port ORACLE,myoracledb.cokm vis0v46q.us-east-1 .rds.amazonaws.com ,ORCL,orcl,orcl123 4,orcl,1521,ORCL,P OSTGRESQL,mypgdbin stance.cokmvis0v46 q.us-east-1.rds.am azonaws.com,pguser ,pgpassword,pgdb,5432 4. Modifica i parametri di configurazione di AWS SCT in base alle tue esigenze, quindi copia il file degli script SQL nella tua directory di lavoro nella `input` sottodirectory.	

Operazione	Description	Competenze richieste
Eseguire lo script Python.	1. Esegui lo script Python usando il seguente comando: `$ python run_aws_sct_sql.py database_migration.txt` 2. Questo crea il file SQL degli oggetti DB. I codici non convertiti con errori di trasformazione possono essere convertiti manualmente.	DBA
Crea gli oggetti in Amazon RDS for PostgreSQL	Esegui i file SQL e crea oggetti nella tua istanza database Amazon RDS for PostgreSQL.	DBA

Risorse correlate

- Oracle su Amazon RDS
- PostgreSQL su Amazon RDS
- Utilizzo dell'interfaccia utente AWS SCT
- Utilizzo di Oracle come sorgente per AWS SCT

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Carica i file BLOB in formato TEXT utilizzando la codifica dei file in Aurora, compatibile con PostgreSQL

Bhanu Ganesh Gudivada e Jeevan Shetty, Amazon Web Services

Riepilogo

Spesso durante la migrazione, ci sono casi in cui è necessario elaborare dati strutturati e non strutturati caricati da file su un file system locale. I dati potrebbero anche essere in un set di caratteri diverso dal set di caratteri del database.

Questi file contengono i seguenti tipi di dati:

- Metadati: questi dati descrivono la struttura del file.
- Dati semistrutturati: si tratta di stringhe di testo in un formato specifico, come JSON o XML. Potresti essere in grado di fare affermazioni su tali dati, ad esempio «inizierà sempre con '<'" o «non contiene caratteri di nuova riga».
- Testo completo: questi dati in genere contengono tutti i tipi di caratteri, inclusi i caratteri di nuova riga e le virgolette. Potrebbe anche essere costituito da caratteri multibyte in UTF-8.
- Dati binari: questi dati possono contenere byte o combinazioni di byte, inclusi valori null e marcatori. end-of-file

Caricare una combinazione di questi tipi di dati può essere una sfida.

Il modello può essere utilizzato con database Oracle locali, database Oracle che si trovano su istanze Amazon Elastic Compute Cloud (Amazon EC2) sul cloud Amazon Web Services (AWS) e Amazon Relational Database Service (Amazon RDS) per database Oracle. Ad esempio, questo modello utilizza Amazon Aurora PostgreSQL Compatible Edition.

In Oracle Database, con l'aiuto di un puntatore `BFILE` (file binario), del `DBMS_LOB` pacchetto e delle funzioni di sistema Oracle, è possibile caricare da file e convertirlo in CLOB con codifica dei caratteri. Poiché PostgreSQL non supporta il tipo di dati BLOB durante la migrazione a un database Edition compatibile con Amazon Aurora PostgreSQL, queste funzioni devono essere convertite in script compatibili con PostgreSQL.

Questo modello fornisce due approcci per caricare un file in una singola colonna di database in un database compatibile con Amazon Aurora PostgreSQL:

- Approccio 1: importi i dati dal tuo bucket Amazon Simple Storage Service (Amazon S3) utilizzando `table_import_from_s3` la funzione dell'estensione con l'opzione `aws_s3` encode.
- Approccio 2: si codifica in formato esadecimale all'esterno del database, quindi si decodifica per visualizzare all'interno del database. `TEXT`

Si consiglia di utilizzare Approach 1 perché Aurora PostgreSQL Compatible ha un'integrazione diretta con l'estensione. `aws_s3`

Questo modello utilizza l'esempio del caricamento di un file flat contenente un modello di e-mail, con caratteri multibyte e una formattazione distinta, in un database compatibile con Amazon Aurora PostgreSQL.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un'istanza Amazon RDS o un'istanza Aurora compatibile con PostgreSQL
- Una conoscenza di base di SQL e Relational Database Management System (RDBMS)
- Un bucket Amazon Simple Storage Service (Amazon S3).
- Conoscenza delle funzioni di sistema in Oracle e PostgreSQL
- Pacchetto RPM HexDump -XXD-0.1.1 (incluso con Amazon Linux 2)

> **Note**
>
> Amazon Linux 2 sta per terminare il supporto. Per ulteriori informazioni, consulta Amazon Linux 2 FAQs.

Limitazioni

- Per il tipo di `TEXT` dati, la stringa di caratteri più lunga possibile che può essere memorizzata è di circa 1 GB.

Versioni del prodotto

- Aurora supporta le versioni di PostgreSQL elencate negli aggiornamenti di Amazon Aurora PostgreSQL.

Architecture

Stack tecnologico Target

- Compatibile con Aurora PostgreSQL

Architettura Target

Approccio 1: utilizzo di aws_s3.table_import_from_s3

Da un server locale, un file contenente un modello di e-mail con caratteri multibyte e formattazione personalizzata viene trasferito su Amazon S3. La funzione di database personalizzata fornita da questo modello utilizza la `aws_s3.table_import_from_s3` funzione with `file_encoding` per caricare file nel database e restituire i risultati delle query come tipo di dati. `TEXT`

1. I file vengono trasferiti nel bucket S3 di staging.
2. I file vengono caricati nel database compatibile con Amazon Aurora PostgreSQL.
3. Utilizzando il client pgAdmin, la `load_file_into_clob` funzione personalizzata viene distribuita nel database Aurora.
4. La funzione personalizzata utilizza internamente file_encoding. `table_import_from_s3` L'output della funzione viene ottenuto utilizzando `array_to_string` e come output. `array_agg` `TEXT`

Approccio 2: codifica in formato esadecimale all'esterno del database e decodifica per visualizzare il TESTO all'interno del database

Un file proveniente da un server locale o da un file system locale viene convertito in un dump esadecimale. Quindi il file viene importato in PostgreSQL come campo. `TEXT`

1. Converti il file in un dump esadecimale nella riga di comando utilizzando l'opzione. `xxd -p`
2. Carica i file di dump esadecimali in Aurora PostgreSQL compatibile utilizzando `\copy` l'opzione, quindi decodifica i file di dump esadecimali in formato binario.

3. TEXTCodifica i dati binari per restituirli come.

Tools (Strumenti)

Servizi AWS

- Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.

Altri strumenti

- pgAdmin4 è una piattaforma di amministrazione e sviluppo open source per PostgreSQL. pgAdmin4 può essere utilizzato su Linux, Unix, mac OS e Windows per gestire PostgreSQL.

Epiche

Approccio 1: importazione di dati da Amazon S3 ad Aurora, compatibile con PostgreSQL

Operazione	Description	Competenze richieste
Avvia un' EC2 istanza.	Per istruzioni sull'avvio di un'istanza, consulta Launch your istance.	DBA
Installa lo strumento pgAdmin del client PostgreSQL.	Scarica e installa pgAdmin.	DBA
Creare una policy IAM	Crea una policy AWS Identity and Access Management (IAM) denominata `aurora-s3-access-pol` che garantisca l'accesso al bucket S3 in cui verranno archiviati i file. Usa il codice seguente, sostituendolo <bucket-n	DBA

Operazione	Description	Competenze richieste

	ame> con il nome del tuo bucket S3.
	
	{
	"Version": "2012-10-17",
	"Statement": [
	{
	"Effect": "Allow",
	"Action": [
	"s3:GetObject",
	"s3:AbortMultipartUpload",
	"s3:DeleteObject",
	"s3:ListMultipartUploadParts",
	"s3:PutObject",
	"s3:ListBucket"
	],
	"Resource": [
	"arn:aws:s3:::<bucket-name>/*",
	"arn:aws:s3:::<bucket-name>"
	]
	}
	]
	}	

Operazione	Description	Competenze richieste
Crea un ruolo IAM per l'importazione di oggetti da Amazon S3 ad Aurora, compatibile con PostgreSQL.	Utilizza il codice seguente per creare un ruolo IAM denominato con la relazione di trust. `aurora-s3-import-role` AssumeRole `AssumeRole` consente ad Aurora di accedere ad altri servizi AWS per tuo conto. `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow","Principal": { "Service": "rds.amazonaws.com" },"Action": "sts:AssumeRole" } ] }`	DBA

Operazione	Description	Competenze richieste
Associa il ruolo IAM al cluster.	Per associare il ruolo IAM al cluster di database compatibile con Aurora PostgreSQL, esegui il seguente comando AWS CLI. Passa `<Account-ID>` all'ID dell'account AWS che ospita il database Aurora compatibile con PostgreSQL. Ciò consente al database compatibile con Aurora PostgreSQL di accedere al bucket S3. `aws rds add-role-to-db-cluster --db-cluster-identifier aurora-postgres-cl --feature-name s3Import --role-arn arn:aws:iam::<Account-ID>:role/aurora-s3-import-role`	DBA
Carica l'esempio su Amazon S3.	1. Nella sezione Informazioni aggiuntive di questo modello, copia il codice del modello di e-mail in un file denominato `salary.event.notification.email.vm` . 2. Carica il file nel bucket S3.	DBA, proprietario dell'app

Operazione	Description	Competenze richieste
Implementa la funzione personalizzata.	1. Dalla sezione Informazioni aggiuntive, copia il contenuto del file `load_file_into_clob` SQL della funzione personalizzata in una tabella temporanea. 2. Accedi al database Aurora compatibile con PostgreSQL e distribuiscilo nello schema del database utilizzando il client pGAdmin.	Proprietario dell'app, DBA

Operazione	Description	Competenze richieste

Esegui la funzione personalizzata per importare i dati nel database.	Esegui il seguente comando SQL, sostituendo gli elementi tra parentesi angolari con i valori appropriati.
	`select load_file _into_clob('aws-s3 -import-test'::tex t,'us-west-1'::tex t,'employee.salary .event.notificatio n.email.vm'::text);`
	Sostituite gli elementi tra parentesi angolari con i valori appropriati, come illustrato nell'esempio seguente, prima di eseguire il comando.
	`Select load_file _into_clob('aws-s3 -import-test'::tex t,'us-west-1'::tex t,'employee.salary .event.notificatio n.email.vm'::text);`
	Il comando carica il file da Amazon S3 e restituisce l'output come. TEXT
		Proprietario dell'app, DBA

Approccio 2: convertire il file modello in un dump esadecimale in un sistema Linux locale

Operazione	Description	Competenze richieste
Converti il file modello in un dump esadecimale.	**Note** L'utilità Hexdump visualizza il contenuto dei file binari in formato esadecimale, decimale, ottale o ASCII. Il `hexdump` comando fa parte del pacchetto e viene preinstallato nelle distribuzioni Linux. `util-linux` Anche il pacchetto RPM Hexdump fa parte di Amazon Linux 2. (: Amazon Linux 2 sta per terminare il supporto. Per ulteriori informazioni, consulta Amazon Linux 2 FAQs.) Per convertire il contenuto del file in un dump esadecimale, esegui il seguente comando shell. `xxd -p </path/file.vm> | tr -d '\n' > </path/file.hex>`	DBA

Operazione	Description	Competenze richieste
	Sostituite il percorso e il file con i valori appropriati, come mostrato nell'esempio seguente. `xxd -p employee.salary.event.notification.email.vm	tr -d '\n' > employee.salary.event.notification.email.vm.hex`	

Operazione	Description	Competenze richieste

Carica il file hexdump nello schema del database.	Usa i seguenti comandi per caricare il file hexdump nel database Aurora compatibile con PostgreSQL.
	
	1. Accedi al database Aurora PostgreSQL e crea una nuova tabella chiamata. `email_template_hex`
	
	`CREATE TABLE email_template_hex(hex_data TEXT);`
	
	2. Caricate i file dal file system locale nello schema del DB utilizzando il seguente comando.
	
	`\copy email_template_hex FROM '/path/file.hex';`
	
	Sostituisci il percorso con la posizione sul file system locale.
	
	`\copy email_template_hex FROM '/tmp/employee.salary.event.notification.email.vm.hex';`
	
	3. Crea un'altra tabella chiamata`email_template_bytea` .
		DBA

Operazione	Description	Competenze richieste

	`CREATE TABLE email_template_bytea(hex_data bytea);`
	4. Inserisci i dati da `email_template_hex` in`email_template_bytea` .
	`INSERT INTO email_template_bytea (hex_data) (SELECT decode(hex_data, 'hex') FROM email_template_hex limit 1);`
	5. Per restituire il codice bytea esadecimale come TEXT dati, esegui il comando seguente.
	`SELECT encode(hex_data::bytea, 'escape') FROM email_template_bytea;`	

Risorse correlate

Riferimenti

- Utilizzo di un database PostgreSQL come destinazione per AWS Database Migration Service
- Playbook sulla migrazione da Oracle Database 19c ad Amazon Aurora con compatibilità PostgreSQL (12.4)

- Creazione di politiche IAM
- Associazione di un ruolo IAM a un cluster Amazon Aurora MySQL DB
- pgAdmin

Tutorial

- Nozioni di base su Amazon RDS
- Esegui la migrazione da Oracle ad Amazon Aurora

Informazioni aggiuntive

funzione personalizzata load_file_into_clob

```
CREATE OR REPLACE FUNCTION load_file_into_clob(
    s3_bucket_name text,
    s3_bucket_region text,
    file_name text,
    file_delimiter character DEFAULT '&'::bpchar,
    file_encoding text DEFAULT 'UTF8'::text)
    RETURNS text
    LANGUAGE 'plpgsql'
    COST 100
    VOLATILE PARALLEL UNSAFE
AS $BODY$
DECLARE
    blob_data BYTEA;
    clob_data TEXT;
    l_table_name CHARACTER VARYING(50) := 'file_upload_hex';
    l_column_name CHARACTER VARYING(50) := 'template';
    l_return_text TEXT;
    l_option_text CHARACTER VARYING(150);
    l_sql_stmt CHARACTER VARYING(500);

BEGIN

    EXECUTE format ('CREATE TEMPORARY TABLE %I (%I text, id_serial serial)',
 l_table_name, l_column_name);

    l_sql_stmt := 'select ''(format text, delimiter ''''' || file_delimiter || ''''',
 encoding ''''' || file_encoding ||  ''''')'' ';
```

```
    EXECUTE FORMAT(l_sql_stmt)
    INTO l_option_text;

    EXECUTE FORMAT('SELECT aws_s3.table_import_from_s3($1,$2,$6,
 aws_commons.create_s3_uri($3,$4,$5))')
    INTO l_return_text
    USING l_table_name, l_column_name, s3_bucket_name,
 file_name,s3_bucket_region,l_option_text;

    EXECUTE format('select array_to_string(array_agg(%I order by id_serial),E''\n'')
 from %I', l_column_name, l_table_name)
    INTO clob_data;

    drop table file_upload_hex;

    RETURN clob_data;
END;
$BODY$;
```

Modello di email

```
##############################################################################################
##
 ##
##    johndoe Template Type: email
 ##
##    File: johndoe.salary.event.notification.email.vm
 ##
##    Author: Aimée Étienne    Date 1/10/2021
      ##
##  Purpose: Email template used by EmplmanagerEJB to inform a johndoe they         ##
##        have been given access to a salary event
 ##
##    Template Attributes:
 ##
##        invitedUser - PersonDetails object for the invited user
  ##
##        salaryEvent - OfferDetails object for the event the user was given access
 ##
##        buyercollege - CompDetails object for the college owning the salary event
 ##
```

```
##        salaryCoordinator - PersonDetails of the salary coordinator for the event
 ##
##        idp - Identity Provider of the email recipient
 ##
##        httpWebRoot - HTTP address of the server
 ##
##
 ##
#################################################################################

$!invitedUser.firstname $!invitedUser.lastname,

Ce courriel confirme que vous avez ete invite par $!salaryCoordinator.firstname $!
salaryCoordinator.lastname de $buyercollege.collegeName a participer a l'evenement
 "$salaryEvent.offeringtitle" sur johndoeMaster Sourcing Intelligence.

Votre nom d'utilisateur est $!invitedUser.username

Veuillez suivre le lien ci-dessous pour acceder a l'evenement.

${httpWebRoot}/myDashboard.do?idp=$!{idp}

Si vous avez oublie votre mot de passe, utilisez le lien "Mot de passe oublie" situe
 sur l'ecran de connexion et entrez votre nom d'utilisateur ci-dessus.

Si vous avez des questions ou des preoccupations, nous vous invitons a
 communiquer avec le coordonnateur de l'evenement $!salaryCoordinator.firstname $!
salaryCoordinator.lastname au ${salaryCoordinator.workphone}.

*******

johndoeMaster Sourcing Intelligence est une plateforme de soumission en ligne pour les
 equipements, les materiaux et les services.

Si vous avez des difficultes ou des questions, envoyez un courriel a
 support@johndoeMaster.com pour obtenir de l'aide.
```

Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL con e utilizzando e AWS SCT AWS DMS AWS CLI CloudFormation

Pinesh Singal, Amazon Web Services

Riepilogo

Questo modello mostra come migrare un'istanza DB Amazon Relational Database Service (Amazon RDS) per Oracle da più terabyte a un'istanza DB Amazon RDS for PostgreSQL utilizzando (). AWS Command Line Interface AWS CLI L'approccio offre tempi di inattività minimi e non richiede l'accesso a. Console di gestione AWS

Questo modello consente di evitare configurazioni manuali e migrazioni individuali utilizzando le console AWS Schema Conversion Tool (AWS SCT) e AWS Database Migration Service (AWS DMS). La soluzione imposta una configurazione unica per più database ed esegue le migrazioni utilizzando e in. AWS SCT AWS DMS AWS CLI

Il modello viene utilizzato AWS SCT per convertire gli oggetti dello schema del database da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL e quindi viene utilizzato per migrare i dati. AWS DMS Utilizzando gli script Python in AWS CLI, crei AWS SCT oggetti e AWS DMS attività con un modello. CloudFormation

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Un'istanza database Amazon RDS for Oracle esistente.
- Un'istanza database Amazon RDS for PostgreSQL esistente.
- Un'istanza Amazon Elastic Compute Cloud (Amazon EC2) o un computer locale con sistema operativo Windows o Linux per l'esecuzione di script.
- Comprensione dei seguenti tipi di attività di AWS DMS migrazione:`full-load`,,`cdc`. `full-load-and-cdc` Per ulteriori informazioni, vedere Creazione di un'attività nella AWS DMS documentazione.
- AWS SCT, installato e configurato con i driver Java Database Connectivity (JDBC) per i motori di database Oracle e PostgreSQL. Per ulteriori informazioni, vedere Installazione e configurazione nella documentazione. AWS SCT AWS SCT

- Il `AWSSchemaConversionToolBatch.jar` file dalla AWS SCT cartella di installazione, copiato nella directory di lavoro.
- Il `cli-sct-dms-cft.zip` file (allegato), scaricato ed estratto nella directory di lavoro.
- La versione più recente del motore di istanze AWS DMS di replica. Per ulteriori informazioni, consulta Come si crea un'istanza di AWS DMS replica nella Supporto AWS documentazione e nelle note di AWS DMS rilascio.
- AWS CLI versione 2, installata e configurata con l'ID della chiave di accesso, la chiave di accesso segreta e Regione AWS il nome predefinito per l' EC2 istanza o il sistema operativo in cui vengono eseguiti gli script. Per ulteriori informazioni, consulta Installazione o aggiornamento alla versione più recente di AWS CLI e Configurazione delle impostazioni contenute AWS CLI nella AWS CLI documentazione.
- Familiarità con CloudFormation i modelli. Per ulteriori informazioni, consulta How CloudFormation works nella CloudFormation documentazione.
- Python versione 3, installato e configurato sull' EC2 istanza o sul sistema operativo in cui vengono eseguiti gli script. Per ulteriori informazioni, consulta la documentazione di Python.

Limitazioni

- I requisiti minimi per l'istanza DB di origine di Amazon RDS for Oracle sono:
  - Versioni Oracle 12c (12.1.0.2, 12.2.0.1), 18c (18.0.0.0) e 19c (19.0.0.0) per le edizioni Enterprise, Standard, Standard One e Standard Two.
  - Sebbene Amazon RDS supporti Oracle 18c (18.0.0.0), questa versione è obsoleta perché Oracle non fornisce più patch per 18c dopo tale data. end-of-support Per ulteriori informazioni, consulta Amazon RDS for Oracle nella documentazione di Amazon RDS.
  - Amazon RDS for Oracle 11g non è più supportato.
- I requisiti minimi per l'istanza database Amazon RDS for PostgreSQL di destinazione sono:
  - PostgreSQL versioni 9 (9.5 e 9.6), 10.x, 11.x, 12.x e 13.x

Versioni del prodotto

- Istanza database Amazon RDS for Oracle versione 12.1.0.2 e successive
- Istanza database Amazon RDS for PostgreSQL versione 11.5 e successive
- AWS CLI versione 2
- L'ultima versione di AWS SCT

- L'ultima versione di Python 3

Architecture

Stack tecnologico di origine

- Amazon RDS per Oracle

Stack tecnologico di destinazione

- Amazon RDS per PostgreSQL

Architettura di origine e destinazione

Il diagramma seguente mostra la migrazione di un'istanza DB Amazon RDS for Oracle verso un'istanza DB Amazon RDS for PostgreSQL utilizzando script Python. AWS DMS

Il diagramma mostra il seguente flusso di lavoro di migrazione:

1. Lo script Python utilizza AWS SCT per connettersi alle istanze DB di origine e di destinazione.
2. L'utente inizia AWS SCT con lo script Python, converte il codice Oracle in codice PostgreSQL e lo esegue sull'istanza DB di destinazione.
3. Lo script Python crea attività di AWS DMS replica per le istanze DB di origine e di destinazione.
4. L'utente distribuisce script Python per avviare le attività e quindi interrompe AWS DMS le attività al termine della migrazione dei dati.

Automazione e scalabilità

Puoi automatizzare questa migrazione aggiungendo parametri e modifiche relative alla sicurezza allo script Python, per fornire funzionalità aggiuntive.

Tools (Strumenti)

- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella shell della riga di comando.

- CloudFormationti aiuta a configurare le risorse AWS, a fornirle in modo rapido e coerente e a gestirle durante tutto il loro ciclo di vita in tutte Account AWS le regioni. Questo modello converte il file `.csv` di input in un file `.json` di input utilizzando uno script Python. Il `.json` file viene utilizzato nei AWS CLI comandi per creare uno CloudFormation stack che crea più attività di AWS DMS replica con Amazon Resource Names (ARNs), tipi di migrazione, impostazioni delle attività e mappature delle tabelle.
- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati verso Cloud AWS o tra combinazioni di configurazioni cloud e locali. Questo modello viene utilizzato AWS DMS per creare, avviare e interrompere le attività con uno script Python eseguito sulla riga di comando e per creare il CloudFormation modello.
- AWS Schema Conversion Tool (AWS SCT) supporta migrazioni di database eterogenee convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione. Questo modello richiede il `AWSSchemaConversionToolBatch.jar` file dalla directory installata. AWS SCT

Codice

Il `cli-sct-dms-cft.zip` file (allegato) contiene il codice sorgente completo per questo pattern.

Epiche

Configura AWS SCT e crea oggetti di database in AWS CLI

Operazione	Description	Competenze richieste
Configura AWS SCT per l'esecuzione da AWS CLI.	1. Configurate i dettagli di configurazione dell'ambiente di origine e di destinazione nel `database_migration.txt` file utilizzando il seguente formato: `#source_vendor,source_hostname,source_dbname,source_user,source_pwd,source_schema,source_p`	DBA

Operazione	Description	Competenze richieste
	`ort,source_sid,tar get_vendor,target_ hostname,target_us er,target_pwd,targ et_dbname,target_p ort ORACLE,myoracl edb.cokmvis0v46q.u s-east-1.rds.amazo naws.com,ORCL,orcl ,orcl1234,orcl,152 1,ORCL,POSTGRESQL, mypgdbinstance.cok mvis0v46q.us-east- 1.rds.amazonaws.co m,pguser,pgpasswor d,pgdb,5432` 2. Modificate i parametri di AWS SCT configura zione in base alle vostre esigenze nei seguenti file:`project_s ettings.x ml` ,`Oracle_PG _Test_Batch.xml` , `eORACLE-orcl-to-POS TGRESQL.xml` .	

Operazione	Description	Competenze richieste
Esegui lo `run_aws_sct.py` script Python.	Esegui lo script `run_aws_sct.py` Python usando il seguente comando: `$ python run_aws_sct.py database_migration.txt` Lo script Python converte gli oggetti del database da Oracle a PostgreSQL e crea file SQL in formato PostgreSQL. Lo script crea anche il file PDF`Database migration assessment report` , che fornisce consigli dettagliati e statistiche di conversione per gli oggetti del database.	DBA
Crea oggetti in Amazon RDS for PostgreSQL.	1. Modifica manualmente i file SQL generati da AWS SCT, se necessario. 2. Esegui i file SQL e crea oggetti nella tua istanza database Amazon RDS for PostgreSQL.	DBA

Configura e crea AWS DMS attività utilizzando e AWS CLI CloudFormation

Operazione	Description	Competenze richieste
Crea un'istanza di AWS DMS replica.	Accedi a Console di gestione AWS, apri la AWS DMS console e crea un'istanza di	DBA

Operazione	Description	Competenze richieste
	replica configurata in base ai tuoi requisiti. Per ulteriori informazioni, consulta Creazione di un'istanza di replica nella AWS DMS documentazione e Come posso creare un'istanza di AWS DMS replica nella documentazione. Supporto AWS	
Crea l'endpoint di origine.	Sulla AWS DMS console, scegli Endpoints, quindi crea un endpoint di origine per il database Oracle in base alle tue esigenze. **Note** L'attributo di connessione aggiuntivo deve avere `numberDataTypeScale` un `-2` valore. Per ulteriori informazioni, consulta Creazione degli endpoint di origine e destinazione nella AWS DMS documentazione.	DBA

Operazione	Description	Competenze richieste
Crea l'endpoint di destinazione.	Sulla AWS DMS console, scegli Endpoints e quindi crea un endpoint di destinazione per il database PostgreSQL in base alle tue esigenze. Per ulteriori informazioni, consulta Creazione di endpoint di origine e destinazione nella documentazione. AWS DMS	DevOps ingegnere

Configura i dettagli AWS DMS della replica da eseguire da. AWS CLI	Configurare gli endpoint di AWS DMS origine e di destinazione e i dettagli di replica nel `dms-arn-list.txt` file con l'ARN dell'endpoint di origine, l'ARN dell'endpoint di destinazione e l'ARN dell'istanza di replica utilizzando il seguente formato:
	
	`#sourceARN,targetARN,repARN`
	`arn:aws:dms:us-east-1:123456789012:endpoint:EH7AINRUDZ5GOYIY6HVMXECMCQ`
	`arn:aws:dms:us-east-1:123456789012:endpoint:HHJVUV57N7O3CQF4PJZKGIOYY5`
	`arn:aws:dms:us-east-1:123456789012:rep:LL57N77AQQAHHJF4PJFHNEDZ5G`
		DBA

Operazione	Description	Competenze richieste
Esegui lo script `dms-create-task.py` Python per creare le AWS DMS attività.	1. Esegui lo script `dms-create-task.py` Python usando il seguente comando: `$ python dms-create-task.py database_migration.txt dms-arn-list.txt <cft-stack-name> <migration-type>` dove: • `database_migration.txt` è il file di testo per la migrazione del database. • `dms-arn-list.txt` è l'elenco ARN per. AWS DMS • `<cft-stack-name>` è il nome dello CloudFormation stack definito dall'utente. • `<migration-type>` è `full-load` , `cdc` o `full-load-and-cdc` . 2. A seconda del tipo di migrazione, è possibile utilizzare i seguenti comandi per creare tre tipi di AWS DMS attività: • `$ python dms-create-task.py`	DBA

Operazione	Description	Competenze richieste
	`database_migration.txt dms-arn-list.txt dms-cli-cft-stack full-load` • `$ python dms-create-task.py database_migration.txt dms-arn-list.txt dms-cli-cft-stack cdc` • `$ python dms-create-task.py database_migration.txt dms-arn-list.txt dms-cli-cft-stack full-load-and-cdc`	
Verifica che AWS DMS le attività siano pronte.	Sulla AWS DMS console, verifica che AWS DMS le tue attività siano in `Ready` stato nella sezione Stato.	DBA

Avvia e interrompi le AWS DMS attività utilizzando AWS CLI

Operazione	Description	Competenze richieste
Avvia le AWS DMS attività.	Esegui lo script `dms-start-task.py` Python usando il seguente comando:	DBA

Operazione	Description	Competenze richieste
	`$ python dms-start-task.py start '<cdc-start-datetime>'` **Note** La data e l'ora di inizio devono essere nel `'YYYY-MM-DDTHH:MI:SS'` formato `'DD-MON-YYYY'` o (ad esempio, `'01-Dec-2019'` o `'2018-03-08T12:12:12'`). È possibile esaminare lo stato delle AWS DMS attività nella scheda Statistiche della tabella nella pagina Attività della AWS DMS console.	

Operazione	Description	Competenze richieste
Convalida i dati.	1. Una volta completata la migrazione a pieno carico, l'attività viene mantenuta continuamente in esecuzione per CDC. 2. Quando il CDC è completo o non è necessario migrare altre modifiche, rivedi e convalida i risultati e i dati delle attività di migrazione nei database Oracle e PostgreSQL. È possibile convalidare i dati controllando lo stato e il conteggio delle colonne (**`Validation state`**, **`Validation pending`** **`Validation failed`** **`Validation suspended`** , e**`Validation details`**) nella scheda Statistiche delle tabelle del task di migrazione del database nella pagina Attività della console. AWS DMS Per ulteriori informazioni, consulta la convalida AWS DMS dei dati nella AWS DMS documentazione.	DBA

Operazione	Description	Competenze richieste
Interrompi le AWS DMS attività.	Esegui lo script Python usando il seguente comando: `$ python dms-start-task.py stop` **Note** AWS DMS le attività potrebbero interromp ersi con uno `failed` stato, a seconda dello stato di convalida. Per ulteriori informazioni, consulta la sezione successiva	DBA

risoluzione dei problemi

Problema	Soluzione
AWS SCT le connessioni di test di origine e destinazione falliscono.	Configura le versioni dei driver JDBC e le regole in entrata del gruppo di sicurezza VPC per accettare il traffico in entrata.
L'esecuzione del test dell'endpoint di origine o di destinazione non riesce.	Controlla se le impostazioni dell'endpoint e l'istanza di replica sono in stato. `Available` Controlla se lo stato della connessione dell'endpoint è. `Successful` Per ulteriori informazioni, consulta Come posso risolvere gli errori di connettività degli endpoint AWS DMS nella documentazione. Supporto AWS

Problema	Soluzione
L'esecuzione a pieno carico non riesce.	Controlla se i database di origine e di destinazione hanno tipi e dimensioni di dati corrispondenti. Per ulteriori informazioni, consulta Risoluzione dei problemi di migrazione AWS DMS nella AWS DMS documentazione.
Si verificano errori di esecuzione della convalida.	Controlla se la tabella ha una chiave primaria perché le tabelle a chiave non primaria non vengono convalidate. Se la tabella ha una chiave primaria e sono presenti degli errori, verifica che l'attributo di connessione aggiuntivo nell'endpoint di origine lo abbia. `numberDataTypeScale=-2` Per ulteriori informazioni, consulta Impostazioni degli endpoint quando si utilizza Oracle come fonte e Risoluzione dei problemi nella AWS DMS documentazione. AWS DMSOracleSettings

Risorse correlate

- Installazione e configurazione AWS SCT
- Introduzione a AWS DMS (video)
- Esempi di comandi di funzionamento CloudFormation dello stack per e AWS CLI PowerShell
- Navigazione nell'interfaccia utente di AWS SCT
- Utilizzo di un database Oracle come fonte per AWS DMS
- Connessione ai database Oracle con AWS SCT
- Utilizzo di un database PostgreSQL come destinazione per AWS DMS
- Fonti per la migrazione dei dati
- Obiettivi per la migrazione dei dati

- cloud formation (documentazione)AWS CLI
- create-stack (documentazione)AWS CLI
- dms (documentazione)AWS CLI

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL in modalità SSL utilizzando AWS DMS

Pinesh Singal, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni per la migrazione di un'istanza di database Amazon Relational Database Service (Amazon RDS) per Oracle a un database Amazon RDS for PostgreSQL sul cloud Amazon Web Services (AWS). Per crittografare le connessioni tra i database, il modello utilizza l'autorità di certificazione (CA) e la modalità SSL in Amazon RDS e AWS Database Migration Service (AWS DMS).

Il modello descrive una strategia di migrazione online con tempi di inattività minimi o nulli per un database di origine Oracle da più terabyte con un numero elevato di transazioni. Per la sicurezza dei dati, il pattern utilizza SSL per il trasferimento dei dati.

Questo modello utilizza AWS Schema Conversion Tool (AWS SCT) per convertire lo schema del database Amazon RDS for Oracle in uno schema Amazon RDS for PostgreSQL. Quindi il modello utilizza AWS DMS per migrare i dati dal database Amazon RDS for Oracle al database Amazon RDS for PostgreSQL.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Autorità di certificazione (CA) del database Amazon RDS configurata solo con rds-ca-rsa2048-g1
  - Il certificato rds-ca-2019 è scaduto nell'agosto 2024.
  - Il certificato rds-ca-2015 è scaduto il 5 marzo 2020.
- AWS SCT
- AWS DMS
- pgAdmin
- Strumenti SQL (ad esempio, SQL Developer o SQL*Plus)

Limitazioni

- Database Amazon RDS for Oracle: il requisito minimo è per le versioni Oracle 19c per le edizioni Enterprise e Standard Two.
- Database Amazon RDS per PostgreSQL: il requisito minimo è PostgreSQL versione 12 e successive (per le versioni 9.x e successive).

Versioni del prodotto

- Istanza del database Amazon RDS per Oracle versione 12.1.0.2
- Istanza del database Amazon RDS per PostgreSQL versione 11.5

Architecture

Stack tecnologico di origine

- Un'istanza di database Amazon RDS for Oracle con versione 12.1.0.2.v18.

Stack tecnologico Target

- AWS DMS
- Un'istanza di database Amazon RDS for PostgreSQL con versione 11.5.

Architettura Target

Il diagramma seguente mostra l'architettura per l'architettura di migrazione dei dati tra i database Oracle (source) e PostgreSQL (target). L'architettura include quanto segue:

- Un cloud privato virtuale (VPC)
- Una zona di disponibilità
- Una sottorete privata
- Un database Amazon RDS per Oracle
- Un'istanza di replica AWS DMS
- Un database RDS per PostgreSQL

Per crittografare le connessioni per i database di origine e destinazione, è necessario abilitare la modalità CA e SSL in Amazon RDS e AWS DMS.

Tools (Strumenti)

Servizi AWS

- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali.
- Amazon Relational Database Service (Amazon RDS) per Oracle ti aiuta a configurare, gestire e scalare un database relazionale Oracle nel cloud AWS.
- Amazon Relational Database Service (Amazon RDS) per PostgreSQL ti aiuta a configurare, gestire e scalare un database relazionale PostgreSQL nel cloud AWS.
- AWS Schema Conversion Tool (AWS SCT) supporta migrazioni di database eterogenei convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione.

Altri servizi

- pgAdmin è uno strumento di gestione open source per PostgreSQL. Fornisce un'interfaccia grafica che consente di creare, gestire e utilizzare oggetti di database.

Best practice

Amazon RDS fornisce nuovi certificati CA come una best practice di sicurezza AWS. Per informazioni sui nuovi certificati e sulle regioni AWS supportate, consulta Utilizzare SSL/TLS per crittografare una connessione a un'istanza o un cluster DB.

Se la tua istanza RDS utilizza attualmente un certificato `rds-ca-2019` CA e desideri eseguire l'aggiornamento`rds-ca-rsa2048-g1`, segui le istruzioni in Aggiornamento del certificato CA modificando l'istanza o il cluster DB o Aggiornamento del certificato CA applicando la manutenzione.

Epiche

Configurazione dell'istanza Amazon RDS for Oracle

Operazione	Description	Competenze richieste
Creare l'istanza del database Oracle.	Accedi al tuo account AWS, apri la Console di gestione	Informazioni generali su AWS, DBA

Operazione	Description	Competenze richieste
	AWS e accedi alla console Amazon RDS. Sulla console, scegli Crea database, quindi scegli Oracle.	
Configura i gruppi di sicurezza .	Configura i gruppi di sicurezza in entrata e in uscita.	Informazioni generali su AWS
Crea un gruppo di opzioni.	Crea un gruppo di opzioni nello stesso VPC e gruppo di sicurezza del database Amazon RDS for Oracle. Per Opzione, scegli SSL. Per Porta, scegli 2484 (per connessioni SSL).	Informazioni generali su AWS
Configura le impostazioni delle opzioni.	Utilizzare le seguenti impostazioni: • `SQLNET.CIPHER_SUITE` : `SSL_RSA_WITH_AES_256_CBC_SHA` • `SQLNET.SSL_VERSION` : `1.2 or 1.0`	Informazioni generali su AWS
Modifica l'istanza DB RDS for Oracle.	Imposta il certificato CA come rds-ca-rsa2048-g1. In Gruppo di opzioni, allega il gruppo di opzioni creato in precedenza.	DBA, AWS generale

Operazione	Description	Competenze richieste

Verifica che l'istanza DB RDS per Oracle sia disponibile.	Assicurati che l'istanza del database Amazon RDS for Oracle sia attiva e funzionante e che lo schema del database sia accessibile.
	
	Per connetterti a RDS for Oracle DB, usa il `sqlplus` comando dalla riga di comando.
	
	`$ sqlplus orcl/**** @myoracledb.cokmvis0v46q.us-east-1.rds.amazonaws.com:1521/ORCL`
	`SQL*Plus: Release 12.1.0.2.0 Production on Tue Oct 15 18:11:07 2019`
	`Copyright (c) 1982, 2016, Oracle. All rights reserved.`
	`Last Successful login time: Mon Dec 16 2019 23:17:31 +05:30`
	`Connected to:`
	`Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production`
	`With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options`
	`SQL>`
		DBA

Operazione	Description	Competenze richieste
Crea oggetti e dati nel database RDS for Oracle.	Crea oggetti e inserisci dati nello schema.	DBA

Configurazione dell'istanza Amazon RDS per PostgreSQL

Operazione	Description	Competenze richieste
Crea il database RDS per PostgreSQL.	Nella pagina Crea database della console Amazon RDS, scegli PostgreSQL per creare un'istanza di database Amazon RDS for PostgreSQL.	DBA, AWS generale
Configura i gruppi di sicurezza .	Configura i gruppi di sicurezza in entrata e in uscita.	Informazioni generali su AWS
Per creare un gruppo di parametri.	Se utilizzi PostgreSQL versione 11.x, crea un gruppo di parametri per impostare i parametri SSL. Nella versione 12 di PostgreSQL, il gruppo di parametri SSL è abilitato per impostazione predefinita.	Informazioni generali su AWS
Modifica parametri.	Modificate il `rds.force_ssl` parametro in `1` (attivo). Per impostazione predefinita, il `ssl` parametro è `1` (attivo). Impostando il `rds.force_ssl` parametro su`1`, si forza la connessione di tutte le connessioni solo tramite la modalità SSL.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Modifica l'istanza DB RDS per PostgreSQL.	Imposta il certificato CA come 2048-g1. rds-ca-rsa Allega il gruppo di parametri predefinito o il gruppo di parametri creato in precedenza, a seconda della versione di PostgreSQL in uso.	DBA, AWS generale

Operazione	Description	Competenze richieste

Verifica che l'istanza DB RDS per PostgreSQL sia disponibile.	Assicurati che il database Amazon RDS for PostgreSQL sia attivo e funzionante.
	
	Il `psql` comando stabilisce una connessione SSL con `sslmode` set dalla riga di comando.
	
	Un'opzione consiste `sslmode=1` nell'impostare il parametro nel gruppo di parametri e utilizzare una `psql` connessione senza includere il `sslmode` parametro nel comando.
	
	L'output seguente mostra che la connessione SSL è stata stabilita.
	
	`$ psql -h mypgdbinstance.cokmvis0v46q.us-east-1.rds.amazonaws.com -p 5432 "dbname=pgdb user=pguser"`
	`Password for user pguser:`
	`psql (11.3, server 11.5)`
	`SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256, compression: off)`
	`Type "help" for help.`
		DBA

Operazione	Description	Competenze richieste

	`pgdb=>`
	
	Una seconda opzione consiste `sslmode=1` nell'impo stare il gruppo di parametri e includere il `sslmode` parametro nel `psql` comando.
	
	L'output seguente mostra che la connessione SSL è stata stabilita.
	
	`$ psql -h mypgdbins tance.cokmvis0v46q .us-east-1.rds.ama zonaws.com -p 5432 "dbname=pgdb user=pgus er sslmode=require" Password for user pguser: psql (11.3, server 11.5) SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256, compressi on: off) Type "help" for help. pgdb=>`	

Configurazione ed esecuzione di AWS SCT

Operazione	Description	Competenze richieste
Installa AWS SCT.	Installa la versione più recente dell'applicazione AWS SCT.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Configura AWS SCT con i driver JDBC.	Scarica i driver Java Database Connectivity (JDBC) per Oracle (ojdbc8.jar) e PostgreSQL (postgresql-42.2.5.jar). Per configurare i driver in AWS SCT, scegli Impostazioni, Impostazioni globali, Driver.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Crea il progetto AWS SCT.	Crea il progetto e il report AWS SCT, utilizzando Oracle come motore DB di origine e Amazon RDS for PostgreSQL come motore DB di destinazione: 1. Verifica le connessioni al database Oracle di origine e scegli come destinazione il database Amazon RDS for PostgreSQL fornendo i dettagli di connessione. Per il database Oracle di origine, sono richieste le seguenti autorizzazioni o privilegi: • `CONNECT` • `SELECT_CATALOG_ROLE` • `SELECT ANY DICTIONARY` • `SELECT on SYS.USER$ TO <sct_user>` Per ulteriori informazioni, consulta Using Oracle Database as a source for AWS SCT. Sia le connessioni di origine che quelle di destinazione devono avere successo prima che AWS SCT possa	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	avviare il report di migrazione. 2. Dopo il report, inserisci lo schema da convertire e scegli Fine.	
Convalida gli oggetti del database.	1. Scegliete Load schema. AWS SCT visualizza l'origine e gli oggetti di destinazione convertiti, inclusi gli oggetti che presentano errori. Aggiorna eventuali oggetti errati nel database di destinazione. 2. Esamina gli errori e cancellali utilizzando l'intervento manuale. 3. Dopo aver eliminato tutti gli errori, scegli nuovamente Carica schema. 4. Scegli Applica al database. 5. Connettiti a pgAdmin o a qualsiasi strumento che supporti una connessione DB PostgreSQL e controlla lo schema e gli oggetti.	DBA, AWS generale

Configurazione ed esecuzione di AWS DMS

Operazione	Description	Competenze richieste
Creare un'istanza di replica.	1. Accedi al tuo account, apri la Console di gestione AWS	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	e accedi alla console AWS DMS. 2. Crea un'istanza di replica con impostazioni valide per VPC, gruppo di sicurezza , zona di disponibilità e attributi di connessione aggiuntivi.	
Importa il certificato.	Scarica il pacchetto di certificati (PEM) per la tua regione AWS. Il pacchetto contiene sia i certificati `rds-ca-2019` intermedi che quelli root. Il pacchetto contiene anche i certificati `rds-ca-rsa2048-g1` CA `rds-ca-rsa4096-g1` e `rds-ca-ecc384-g1` root. L'application trust store deve registrare solo il certificato CA principale.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Crea l'endpoint di origine.	1. Crea un endpoint di origine per Amazon RDS for Oracle scegliendo Seleziona istanza DB RDS e quindi selezionando l'istanza DB RDS per Oracle che hai creato. I dettagli di configurazione dell'endpoint verranno compilati automaticamente. 2. Scegli Fornisci le informazioni di accesso manualmente. Per Port, assicurati di inserire 2484. 3. In modalità Secure Socket Layer (SSL)`verify-ca`, scegli, quindi scegli il certificato CA creato in precedenza. 4. In Impostazioni Endpoint, aggiungi l'attributo di connessione aggiuntivo `NumberDataTypeScale=-2` per supportare il tipo di `NUMBER` dati senza dimensioni. Per ulteriori informazioni, consulta Usare un database Oracle come sorgente per AWS Database Migration Service.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Crea l'endpoint di destinazione.	1. Crea un endpoint di destinazione per Amazon RDS for PostgreSQL scegliendo Seleziona istanza DB RDS e quindi selezionando l'istanza DB RDS per PostgreSQL. I dettagli di configurazione dell'endpoint verranno compilati automaticamente. 2. Scegli Fornisci le informazioni di accesso manualmente. Per Port, assicurati di inserire 2484. Per ulteriori informazioni, consulta Usare un database PostgreSQL come destinazione per AWS Database Migration Service.	Informazioni generali su AWS
Testa gli endpoint.	1. Testa gli endpoint di origine e di destinazione per confermare che entrambi abbiano esito positivo e siano disponibili. 2. Se un test fallisce, assicurati che le regole in entrata del gruppo di sicurezza siano valide.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Crea attività di migrazione.	Per creare un'attività di migrazione per l'acquisizione dei dati a pieno carico e modifica (CDC) o per la convalida dei dati, procedi come segue: 1. Per creare un'attività di migrazione del database, scegli l'istanza di replica, l'endpoint del database di origine, l'endpoint del database di destinazione. Specificate il tipo di migrazione come uno dei seguenti: • Migrazione dei dati esistenti (a pieno carico) • Replica solo le modifiche ai dati (CDC) • Migra i dati esistenti e replica le modifiche in corso (a pieno carico e CDC) 2. In Table Mappings, puoi configurare le regole di selezione e le regole di trasformazione nei formati GUI o JSON: • In Regole di selezione, seleziona lo schema, inserisci il nome della tabella e seleziona l'azione (Includi o	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	Escludi) da configura re, ad esempio Schema ORCL, Nome tabella%, Action Include. • In Regole di trasforma zione, effettuate una delle seguenti operazioni: • Selezionate lo schema e scegliete l'azione (maiuscole, prefisso, suffisso), ad esempio Target Schema ORCL, Action Make in minuscolo. • Seleziona lo schema, inserisci il nome della tabella e scegli l'azione (maiuscole, prefisso, suffisso), ad esempio Target Schema ORCL, Table%, Action Make in minuscolo. 3. Attiva il monitoraggio di Amazon CloudWatch Logs. 4. Per le regole di mappatura, aggiungi il seguente codice JSON. `{ "rules": [ { "rule-type": "transformation",`	

Operazione	Description	Competenze richieste
	"rule-id" : "1", "rule-name": "1", "rule-target": "table", "object-locator": { "schema-name": "%", "table-name": "%" }, "rule-action": "convert-lowercase", "value": null, "old-value": null }, { "rule-type": "transformation", "rule-id" : "2", "rule-name": "2", "rule-target": "schema", "object-locator": { "schema-name": "ORCL", "table-name": "%" }, "rule-action": "convert-lowercase",	

Operazione	Description	Competenze richieste
	"value": null, "old-value": null }, { "rule-type": "selection", "rule-id": "3", "rule-name": "3", "object-locator": { "schema-name": "ORCL", "table-name": "DEPT" }, "rule-action": "include", "filters": [] }] }	
Pianifica il ciclo di produzione.	Conferma i tempi di inattività con le parti interessate, come i proprietari delle applicazioni, per eseguire AWS DMS nei sistemi di produzione.	Responsabile della migrazione

Operazione	Description	Competenze richieste
Esegui l'attività di migrazione di .	1. Avvia l'attività AWS DMS con stato Ready e monitora i log delle attività di migrazione in Amazon CloudWatch per eventuali errori. Se hai scelto Migra i dati esistenti e replica le modifiche in corso come tipo di migrazione e lo stato è Carica replica completa in corso, la migrazione completa dei dati con CDC è completata e la convalida è in corso. 2. Dopo aver avviato la migrazione, puoi ottenere ulteriori informazioni sulla connessione SSL in. CloudWatch Per Oracle, CloudWatch mostra la seguente stringa di connessione. `2019-12-17T09:15:11 [SOURCE_UNLOAD ]I: Connecting to Oracle: Beginning session (oracle_endpoint_conn.c:834)`	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	La stringa di connessione PostgreSQL sarà simile all'esempio seguente. `2019-12-17T09:15:11 [TARGET_LOAD ]I: Going to connect to ODBC connection string: PROTOCOL=7.4-0;DRIVER={PostgreSQL};SERVER=mypgdbinstance.cokmvis0v46q.us-east-1.rds.amazonaws.com;DATABASE=pgdb;PORT=5432;sslmode=require;UID=pguser; (odbc_endpoint_imp.c:2218)`	
Convalida i dati.	Esamina i risultati e i dati delle attività di migrazione nei database Oracle di origine e PostgreSQL di destinazione: 1. Connect a pgAdmin e controlla i dati nel tuo database PostgreSQL con schema. `ORCL` 2. Per CDC, controlla le modifiche in corso inserendo o aggiornando i dati nel database Oracle di origine.	DBA

Operazione	Description	Competenze richieste
Interrompi l'attività di migrazione.	Dopo aver completato con successo la convalida dei dati, interrompi l'attività di migrazione.	Informazioni generali su AWS

Pulisci le risorse

Operazione	Description	Competenze richieste
Elimina le attività di AWS DMS.	1. Sulla console AWS DMS, accedi alle attività di migrazione del database e interrompi qualsiasi attività AWS DMS in corso o in esecuzione. 2. Seleziona l'attività o le attività, scegli Azioni e scegli Elimina.	Informazioni generali su AWS
Elimina gli endpoint AWS DMS.	Seleziona gli endpoint di origine e di destinazione che hai creato, scegli Azioni e scegli Elimina.	Informazioni generali su AWS
Elimina l'istanza di replica AWS DMS.	Scegli l'istanza di replica, scegli Azioni, quindi scegli Elimina.	Informazioni generali su AWS
Eliminare il database PostgreSQL.	1. Sulla console Amazon RDS, scegli Databases. 2. Seleziona l'istanza del database PostgreSQL che hai creato, scegli Azioni, quindi scegli Elimina.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Elimina il database Oracle.	Sulla console Amazon RDS, seleziona l'istanza del database Oracle, scegli Azioni, quindi scegli Elimina.	Informazioni generali su AWS

risoluzione dei problemi

Problema	Soluzione
Le connessioni di test di origine e destinazione di AWS SCT non funzionano.	Configura le versioni dei driver JDBC e le regole in entrata del gruppo di sicurezza VPC per accettare il traffico in entrata.
L'esecuzione del test dell'endpoint Oracle Source non riesce.	Controlla le impostazioni dell'endpoint e se l'istanza di replica è disponibile.
L'esecuzione a pieno carico dell'attività AWS DMS non riesce.	Verifica se i database di origine e di destinazi one hanno tipi e dimensioni di dati corrispon denti.
L'attività di convalida e migrazione di AWS DMS restituisce errori.	1. Verifica se la tabella ha una chiave primaria. Le tabelle senza una chiave primaria non vengono convalidate. 2. Se la tabella ha una chiave primaria ma restituisce errori, controlla l'attributo di connessione aggiuntivo nell'endpoint di origine. L'attributo di connessione aggiuntiv o deve `numberDataTypeScale=-2` supportare il tipo di `NUMBER` dati senza dimensioni in modo dinamico in base ai dati disponibili nella tabella.

Risorse correlate

Database

- Amazon RDS per Oracle
- Amazon RDS per PostgreSQL

Connessione DB SSL

- Utilizzo SSL/TLS per crittografare una connessione a un'istanza DB
  - Utilizzo di SSL con un'istanza RDS per Oracle DB
  - Protezione delle connessioni a RDS per PostgreSQL con SSL/TLS
  - Scarica pacchetti di certificati per regioni AWS specifiche
    - Scarica il certificato principale CA-2019 (scaduto ad agosto 2024)
- Lavorare con i gruppi di opzioni
  - Aggiungere opzioni alle istanze di Oracle DB
  - Oracle Secure Sockets Layer
- Lavorare con gruppi di parametri
- Parametro di connessione PostgreSQL sslmode
- Utilizzo di SSL da JDBC
- Rotazione del certificato SSL/TLS
  - Aggiornamento del certificato CA modificando l'istanza o il cluster di database
  - Aggiornamento del certificato CA applicando la manutenzione

AWS SCT

- Strumento di conversione dello schema AWS
- Guida per l'utente di AWS Schema Conversion Tool
- Utilizzo dell'interfaccia utente AWS SCT
- Utilizzo di Oracle Database come sorgente per AWS SCT

AWS DMS

- AWS Database Migration Service

- Guida per l'utente di AWS Database Migration Service
  - Utilizzo di un database Oracle come origine per AWS DMS
  - Utilizzo di un database PostgreSQL come destinazione per AWS DMS
- Utilizzo di SSL con AWS Database Migration Service
- Migrazione di applicazioni che eseguono database relazionali su AWS

Informazioni aggiuntive

I certificati Amazon RDS Certificate Authority sono `rds-ca-2019` scaduti nell'agosto 2024. Se utilizzi o prevedi di utilizzare SSL o TLS con verifica del certificato per connetterti alle istanze DB RDS o ai cluster DB Multi-AZ, prendi in considerazione l'utilizzo di uno dei nuovi certificati CA:, , o. `rds-ca-rsa2048-g1` `rds-ca-rsa4096-g1` `rds-ca-ecc384-g1`

Migrazione dei pacchetti pragma Oracle SERIALLY_REUSABLE in PostgreSQL

Vinay Paladi, Amazon Web Services

Riepilogo

Questo modello fornisce un step-by-step approccio per la migrazione dei pacchetti Oracle definiti come pragma SERIALLY_REUSABLE su PostgreSQL su Amazon Web Services (AWS). Questo approccio mantiene la funzionalità del pragma SERIALLY_REUSABLE.

PostgreSQL non supporta il concetto di pacchetti e il pragma SERIALLY_REUSABLE. Per ottenere funzionalità simili in PostgreSQL, puoi creare schemi per pacchetti e distribuire tutti gli oggetti correlati (come funzioni, procedure e tipi) all'interno degli schemi. Per ottenere la funzionalità del pragma SERIALLY_REUSABLE, lo script di funzione wrapper di esempio fornito in questo modello utilizza un pacchetto di estensione AWS Schema Conversion Tool (AWS SCT).

Per ulteriori informazioni, consulta SERIALLY_REUSABLE Pragma nella documentazione Oracle.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- La versione più recente di AWS SCT e i driver richiesti
- Un database Amazon Aurora compatibile con PostgreSQL Edition o un database Amazon Relational Database Service (Amazon RDS) per PostgreSQL

Versioni del prodotto

- Database Oracle versione 10g e successive

Architecture

Stack tecnologico di origine

- Database Oracle locale

Stack tecnologico Target

- Compatibile con Aurora PostgreSQL o Amazon RDS per PostgreSQL

- AWS SCT

Architettura di migrazione

Tools (Strumenti)

Servizi AWS

- AWS Schema Conversion Tool (AWS SCT) supporta migrazioni di database eterogenei convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione.
- Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL.
- Amazon Relational Database Service (Amazon RDS) per PostgreSQL ti aiuta a configurare, gestire e scalare un database relazionale PostgreSQL nel cloud AWS.

Altri strumenti

- pgAdmin è uno strumento di gestione open source per PostgreSQL. Fornisce un'interfaccia grafica che consente di creare, gestire e utilizzare oggetti di database.

Epiche

Esegui la migrazione del pacchetto Oracle utilizzando AWS SCT

Operazione	Description	Competenze richieste
Configura AWS SCT.	Configura la connettività AWS SCT al database di origine. Per ulteriori informazioni, consulta Using Oracle Database as a source for AWS SCT.	DBA, Sviluppatore
Converti lo script.	Usa AWS SCT per convertire il pacchetto Oracle seleziona	DBA, Sviluppatore

Operazione	Description	Competenze richieste
	ndo il database di destinazione come compatibile con Aurora PostgreSQL.	
Salva i file.sql.	Prima di salvare il file.sql, modifica l'opzione Project Settings in AWS SCT su File singolo per fase. AWS SCT separerà il file.sql in più file.sql in base al tipo di oggetto.	DBA, Sviluppatore
Cambia il codice.	Apri la `init` funzione generata da AWS SCT e modificala come mostrato nell'esempio nella sezione Informazioni aggiuntive. Aggiungerà una variabile per ottenere la funzionalità. `pg_serialize = 0`	DBA, Sviluppatore
Prova la conversione.	Implementa la `init` funzione nel database Aurora compatibile con PostgreSQL e verifica i risultati.	DBA, Sviluppatore

Risorse correlate

- Strumento di conversione dello schema AWS
- Amazon RDS
- Caratteristiche di Amazon Aurora
- SERIALLY_REUSABLE Pragma

Informazioni aggiuntive

```
Source Oracle Code:
```

```
CREATE OR REPLACE PACKAGE test_pkg_var
IS
PRAGMA SERIALLY_REUSABLE;
PROCEDURE function_1
 (test_id number);
PROCEDURE function_2
 (test_id number
 );
END;

CREATE OR REPLACE PACKAGE BODY test_pkg_var
IS
PRAGMA SERIALLY_REUSABLE;
v_char VARCHAR2(20) := 'shared.airline';
v_num number := 123;

PROCEDURE function_1(test_id number)
IS
begin
dbms_output.put_line( 'v_char-'|| v_char);
dbms_output.put_line( 'v_num-'||v_num);
v_char:='test1';
function_2(0);
END;

PROCEDURE function_2(test_id number)
is
begin
dbms_output.put_line( 'v_char-'|| v_char);
dbms_output.put_line( 'v_num-'||v_num);
END;
END test_pkg_var;

Calling the above functions

set serveroutput on


EXEC test_pkg_var.function_1(1);


EXEC test_pkg_var.function_2(1);
```

```
Target Postgresql Code:


CREATE SCHEMA test_pkg_var;

CREATE OR REPLACE FUNCTION test_pkg_var.init(pg_serialize IN INTEGER DEFAULT 0)

RETURNS void
AS
$BODY$

DECLARE

BEGIN

if aws_oracle_ext.is_package_initialized( 'test_pkg_var' ) AND pg_serialize = 0

then

return;

end if;

PERFORM aws_oracle_ext.set_package_initialized( 'test_pkg_var' );

PERFORM aws_oracle_ext.set_package_variable( 'test_pkg_var', 'v_char',
 'shared.airline.basecurrency'::CHARACTER

VARYING(100));

PERFORM aws_oracle_ext.set_package_variable('test_pkg_var', 'v_num', 123::integer);

END;

$BODY$

LANGUAGE plpgsql;


CREATE OR REPLACE FUNCTION test_pkg_var.function_1(pg_serialize int default 1)

RETURNS void
AS
```

```
$BODY$
DECLARE

BEGIN

PERFORM test_pkg_var.init(pg_serialize);

raise notice 'v_char%',aws_oracle_ext.get_package_variable( 'test_pkg_var', 'v_char');

raise notice 'v_num%',aws_oracle_ext.get_package_variable( 'test_pkg_var', 'v_num');

PERFORM aws_oracle_ext.set_package_variable( 'test_pkg_var', 'v_char',
 'test1'::varchar);

PERFORM test_pkg_var.function_2(0);
END;

$BODY$
LANGUAGE plpgsql;


CREATE OR REPLACE FUNCTION test_pkg_var.function_2(IN pg_serialize integer default 1)

RETURNS void

AS

$BODY$

DECLARE

BEGIN

PERFORM test_pkg_var.init(pg_serialize);

raise notice 'v_char%',aws_oracle_ext.get_package_variable( 'test_pkg_var', 'v_char');

raise notice 'v_num%',aws_oracle_ext.get_package_variable( 'test_pkg_var', 'v_num');

END;
$BODY$
LANGUAGE plpgsql;
```

```
Calling the above functions

select test_pkg_var.function_1()

 select test_pkg_var.function_2()
```

Esegui la migrazione di tabelle esterne Oracle verso Amazon Aurora, compatibile con PostgreSQL

anuradha chintha e Rakesh Raghav, Amazon Web Services

Riepilogo

Le tabelle esterne offrono a Oracle la possibilità di interrogare i dati archiviati all'esterno del database in file flat. È possibile utilizzare il driver ORACLE_LOADER per accedere a qualsiasi dato memorizzato in qualsiasi formato che possa essere caricato dall'utilità SQL*Loader. Non è possibile utilizzare Data Manipulation Language (DML) su tabelle esterne, ma è possibile utilizzare tabelle esterne per operazioni di interrogazione, join e ordinamento.

Amazon Aurora PostgreSQL Compatible Edition non offre funzionalità simili alle tabelle esterne di Oracle. È invece necessario utilizzare la modernizzazione per sviluppare una soluzione scalabile che soddisfi i requisiti funzionali e sia parsimoniosa.

Questo modello fornisce i passaggi per la migrazione di diversi tipi di tabelle esterne Oracle all'edizione compatibile con Aurora PostgreSQL sul cloud Amazon Web Services (AWS) utilizzando l'estensione. `aws_s3`

Consigliamo di testare a fondo questa soluzione prima di implementarla in un ambiente di produzione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Interfaccia a riga di comando di AWS (CLI AWS)
- Un'istanza di database compatibile con Aurora PostgreSQL disponibile.
- Un database Oracle locale con una tabella esterna
- API PG.Client
- File di dati

Limitazioni

- Questo modello non fornisce la funzionalità necessaria per sostituire le tabelle esterne Oracle. Tuttavia, i passaggi e il codice di esempio possono essere ulteriormente migliorati per raggiungere gli obiettivi di modernizzazione del database.
- I file non devono contenere il carattere che viene utilizzato come delimitatore nelle funzioni di `aws_s3` esportazione e importazione.

Versioni del prodotto

- Per importare da Amazon S3 in RDS per PostgreSQL, il database deve eseguire PostgreSQL versione 10.7 o successiva.

Architecture

Stack tecnologico di origine

- Oracle

Architettura di origine

Stack tecnologico Target

- Compatibile con Amazon Aurora PostgreSQL
- Amazon CloudWatch
- AWS Lambda
- AWS Secrets Manager
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Storage Service (Amazon S3)

Architettura Target

Il diagramma seguente mostra una rappresentazione di alto livello della soluzione.

1. I file vengono caricati nel bucket S3.
2. Viene avviata la funzione Lambda.

3. La funzione Lambda avvia la chiamata alla funzione DB.
4. Secrets Manager fornisce le credenziali per l'accesso al database.
5. A seconda della funzione DB, viene creato un allarme SNS.

Automazione e scalabilità

Qualsiasi aggiunta o modifica alle tabelle esterne può essere gestita con la manutenzione dei metadati.

Tools (Strumenti)

- Compatibile con Amazon Aurora PostgreSQL — Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito, compatibile con PostgreSQL e conforme ad ACID che combina la velocità e l'affidabilità dei database commerciali di fascia alta con l'economicità dei database open source.
- AWS CLI — AWS Command Line Interface (AWS CLI) è uno strumento unificato per gestire i servizi AWS. Con un solo strumento da scaricare e configurare, puoi controllare più servizi AWS dalla riga di comando e automatizzarli tramite script.
- Amazon CloudWatch: Amazon CloudWatch monitora le risorse e l'utilizzo di Amazon S3.
- AWS Lambda: AWS Lambda è un servizio di elaborazione serverless che supporta l'esecuzione di codice senza effettuare il provisioning o la gestione di server, creare una logica di scalabilità del cluster in base al carico di lavoro, mantenere integrazioni di eventi o gestire i runtime. In questo modello, Lambda esegue la funzione di database ogni volta che un file viene caricato su Amazon S3.
- AWS Secrets Manager — AWS Secrets Manager è un servizio per l'archiviazione e il recupero delle credenziali. Utilizzando Secrets Manager, puoi sostituire le credenziali codificate nel codice, comprese le password, con una chiamata API a Secrets Manager per recuperare il segreto a livello di codice.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) fornisce un livello di storage per ricevere e archiviare file per il consumo e la trasmissione da e verso il cluster Aurora compatibile con PostgreSQL.
- aws_s3 — L'estensione `aws_s3` integra la compatibilità con Amazon S3 e Aurora PostgreSQL.
- Amazon SNS — Amazon Simple Notification Service (Amazon SNS) coordina e gestisce la consegna o l'invio di messaggi tra editori e clienti. In questo modello, Amazon SNS viene utilizzato per inviare notifiche.

Codice

Ogni volta che un file viene inserito nel bucket S3, è necessario creare e richiamare una funzione DB dall'applicazione di elaborazione o dalla funzione Lambda. Per i dettagli, consulta il codice (allegato).

Epiche

Crea un file esterno

Operazione	Description	Competenze richieste
Aggiungere un file esterno al database di origine.	Crea un file esterno e spostalo nella `oracle` directory.	DBA

Configurare la destinazione (compatibile con Aurora PostgreSQL)

Operazione	Description	Competenze richieste
Crea un database Aurora PostgreSQL.	Crea un'istanza DB nel tuo cluster compatibile con Amazon Aurora PostgreSQL.	DBA
Crea uno schema, un'estensione aws_s3 e tabelle.	Usa il codice riportato *`ext_tbl_scripts`* nella sezione Informazioni aggiuntive. Le tabelle includono tabelle effettive, tabelle intermedie, tabelle di errore e di registro e una metatabella.	DBA, Sviluppatore
Crea la funzione DB.	Per creare la funzione DB, utilizzate il codice sotto `load_external_table_latest` la funzione nella sezione Informazioni aggiuntive.	DBA, Sviluppatore

Creazione e configurazione della funzione Lambda

Operazione	Description	Competenze richieste
Creare un ruolo.	Crea un ruolo con autorizzazioni per accedere ad Amazon S3 e Amazon Relational Database Service (Amazon RDS). Questo ruolo verrà assegnato a Lambda per l'esecuzione del pattern.	DBA
Creazione della funzione Lambda	Crea una funzione Lambda che legga il nome del file da Amazon S3 (ad esempio`file_key = info.get('object', {}).get('key') )`e chiami la funzione DB (ad esempio`curs.callproc("load_external_tables", [file_key]) )` con il nome del file come parametro di input. A seconda del risultato della chiamata alla funzione, verrà avviata una notifica SNS (ad esempio,). `client.publish(TopicArn='arn:',Message='fileloadsuccess',Subject='fileloadsuccess')` In base alle esigenze aziendali, è possibile creare una	DBA

Operazione	Description	Competenze richieste
	funzione Lambda con codice aggiuntivo, se necessario. Per ulteriori informazioni, consulta la documentazione di Lambda.	
Configura un trigger di evento del bucket S3.	Configura un meccanismo per chiamare la funzione Lambda per tutti gli eventi di creazione di oggetti nel bucket S3.	DBA
Crea un segreto.	Crea un nome segreto per le credenziali del database utilizzando Secrets Manager. Passa il segreto nella funzione Lambda.	DBA
Carica i file di supporto Lambda.	Carica un file.zip che contiene i pacchetti di supporto Lambda e lo script Python allegato per la connessione a Aurora PostgreSQL compatibile. Il codice Python richiama la funzione che hai creato nel database.	DBA
Creare un argomento SNS.	Crea un argomento SNS per inviare posta in caso di successo o fallimento del caricamento dei dati.	DBA

Aggiungi l'integrazione con Amazon S3

Operazione	Description	Competenze richieste
Crea un bucket S3.	Sulla console Amazon S3, crea un bucket S3 con un nome univoco che non contenga barre iniziali. Il nome di un bucket S3 è unico a livello globale e lo spazio dei nomi è condiviso da tutti gli account AWS.	DBA
Crea politiche IAM.	Per creare le policy di AWS Identity and Access Management (IAM), usa il codice `s3bucketp olicy_for_import` nella sezione Informazioni aggiuntive.	DBA
Crea ruoli.	Crea due ruoli per la compatibilità con Aurora PostgreSQL, un ruolo per l'importazione e un ruolo per l'esportazione. Assegna le politiche corrispondenti ai ruoli.	DBA
Collega i ruoli al cluster compatibile con Aurora PostgreSQL.	In Gestisci ruoli, collega i ruoli di importazione ed esportazione al cluster Aurora PostgreSQL.	DBA
Crea oggetti di supporto compatibili con Aurora PostgreSQL.	Per gli script delle tabelle, usa il codice riportato nella sezione Informazioni aggiuntive. *`ext_tbl_scripts`*	DBA

Operazione	Description	Competenze richieste
	Per la funzione personalizzata, usa il codice riportato `load_external_Table_latest` nella sezione Informazioni aggiuntive.	

Elabora un file di test

Operazione	Description	Competenze richieste
Carica un file nel bucket S3.	Per caricare un file di test nel bucket S3, usa la console o il seguente comando nella CLI di AWS. `aws s3 cp /Users/Desktop/ukpost/exttbl/"testing files"/aps s3://s3importtest/inputext/aps` Non appena il file viene caricato, un evento bucket avvia la funzione Lambda, che esegue la funzione compatibile con Aurora PostgreSQL.	DBA
Controlla i dati e i file di registro e di errore.	La funzione compatibile con Aurora PostgreSQL carica i file nella tabella principale e crea `.log` file nel bucket S3. `.bad`	DBA

Operazione	Description	Competenze richieste
Monitora la soluzione.	Nella CloudWatch console Amazon, monitora la funzione Lambda.	DBA

Risorse correlate

- Integrazione con Amazon S3
- Amazon S3
- Utilizzo dell'edizione compatibile con Amazon Aurora PostgreSQL
- AWS Lambda
- Amazon CloudWatch
- AWS Secrets Manager
- Configurazione delle notifiche Amazon SNS

Informazioni aggiuntive

ext_table_scripts

```
CREATE EXTENSION aws_s3 CASCADE;
CREATE TABLE IF NOT EXISTS meta_EXTERNAL_TABLE
(
    table_name_stg character varying(100) ,
    table_name character varying(100)  ,
    col_list character varying(1000)  ,
    data_type character varying(100)  ,
    col_order numeric,
    start_pos numeric,
    end_pos numeric,
    no_position character varying(100)  ,
    date_mask character varying(100)  ,
    delimeter character(1)  ,
    directory character varying(100)  ,
    file_name character varying(100)  ,
    header_exist character varying(5)
);
CREATE TABLE IF NOT EXISTS ext_tbl_stg
```

```
(
    col1 text
);
CREATE TABLE IF NOT EXISTS error_table
(
    error_details text,
    file_name character varying(100),
    processed_time timestamp without time zone
);
CREATE TABLE IF NOT EXISTS log_table
(
    file_name character varying(50) COLLATE pg_catalog."default",
    processed_date timestamp without time zone,
    tot_rec_count numeric,
    proc_rec_count numeric,
    error_rec_count numeric
);
sample insert scripts of meta data:
INSERT INTO meta_EXTERNAL_TABLE (table_name_stg, table_name, col_list, data_type,
 col_order, start_pos, end_pos, no_position, date_mask, delimeter, directory,
 file_name, header_exist) VALUES ('F_EX_APS_TRANSACTIONS_STG', 'F_EX_APS_TRANSACTIONS',
 'source_filename', 'character varying', 2, 8, 27, NULL, NULL, NULL, 'databasedev',
 'externalinterface/loaddir/APS', 'NO');
INSERT INTO meta_EXTERNAL_TABLE (table_name_stg, table_name, col_list, data_type,
 col_order, start_pos, end_pos, no_position, date_mask, delimeter, directory,
 file_name, header_exist) VALUES ('F_EX_APS_TRANSACTIONS_STG', 'F_EX_APS_TRANSACTIONS',
 'record_type_identifier', 'character varying', 3, 28, 30, NULL, NULL, NULL,
 'databasedev', 'externalinterface/loaddir/APS', 'NO');
INSERT INTO meta_EXTERNAL_TABLE (table_name_stg, table_name, col_list, data_type,
 col_order, start_pos, end_pos, no_position, date_mask, delimeter, directory,
 file_name, header_exist) VALUES ('F_EX_APS_TRANSACTIONS_STG', 'F_EX_APS_TRANSACTIONS',
 'fad_code', 'numeric', 4, 31, 36, NULL, NULL, NULL, 'databasedev', 'externalinterface/
loaddir/APS', 'NO');
INSERT INTO meta_EXTERNAL_TABLE (table_name_stg, table_name, col_list, data_type,
 col_order, start_pos, end_pos, no_position, date_mask, delimeter, directory,
 file_name, header_exist) VALUES ('F_EX_APS_TRANSACTIONS_STG', 'F_EX_APS_TRANSACTIONS',
 'session_sequence_number', 'numeric', 5, 37, 42, NULL, NULL, NULL, 'databasedev',
 'externalinterface/loaddir/APS', 'NO');
INSERT INTO meta_EXTERNAL_TABLE (table_name_stg, table_name, col_list, data_type,
 col_order, start_pos, end_pos, no_position, date_mask, delimeter, directory,
 file_name, header_exist) VALUES ('F_EX_APS_TRANSACTIONS_STG', 'F_EX_APS_TRANSACTIONS',
 'transaction_sequence_number', 'numeric', 6, 43, 48, NULL, NULL, NULL, 'databasedev',
 'externalinterface/loaddir/APS', 'NO');
```

s3bucketpolicy_for import

```
---Import role policy
--Create an IAM policy to allow, Get,  and list actions on S3 bucket
 {
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "s3import",
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:s3:::s3importtest",
                "arn:aws:s3:::s3importtest/*"
            ]
        }
    ]
}
--Export Role policy
--Create an IAM policy to allow, put,  and list actions on S3 bucket
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "s3export",
            "Action": [
                "S3:PutObject",
                "s3:ListBucket"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:s3:::s3importtest/*"
            ]
        }
    ]
}
```

Esempio di funzione DB load_external_tables_latest

```
CREATE OR REPLACE FUNCTION public.load_external_tables(pi_filename text)
```

```
 RETURNS character varying
 LANGUAGE plpgsql
AS $function$
/* Loading data from S3 bucket into a APG table */
DECLARE
 v_final_sql TEXT;
 pi_ext_table TEXT;
 r refCURSOR;
 v_sqlerrm text;
 v_chunk numeric;
 i integer;
 v_col_list TEXT;
 v_postion_list CHARACTER VARYING(1000);
 v_len  integer;
 v_delim varchar;
 v_file_name CHARACTER VARYING(1000);
 v_directory CHARACTER VARYING(1000);
 v_table_name_stg CHARACTER VARYING(1000);
 v_sql_col TEXT;
 v_sql TEXT;
 v_sql1 TEXT;
 v_sql2 TEXT;
 v_sql3 TEXT;
 v_cnt integer;
 v_sql_dynamic TEXT;
 v_sql_ins TEXT;
 proc_rec_COUNT integer;
 error_rec_COUNT integer;
 tot_rec_COUNT integer;
 v_rec_val integer;
 rec record;
 v_col_cnt integer;
 kv record;
 v_val text;
 v_header text;
 j integer;
 ERCODE VARCHAR(5);
 v_region text;
 cr CURSOR FOR
 SELECT distinct DELIMETER,
   FILE_NAME,
   DIRECTORY
 FROM  meta_EXTERNAL_TABLE
 WHERE table_name = pi_ext_table
```

```
   AND DELIMETER IS NOT NULL;


 cr1 CURSOR FOR
   SELECT   col_list,
   data_type,
   start_pos,
   END_pos,
   concat_ws('',' ',TABLE_NAME_STG) as TABLE_NAME_STG,
   no_position,date_mask
 FROM  meta_EXTERNAL_TABLE
 WHERE table_name = pi_ext_table
 order by col_order asc;
cr2 cursor FOR
SELECT  distinct table_name,table_name_stg
   FROM  meta_EXTERNAL_TABLE
   WHERE upper(file_name) = upper(pi_filename);


BEGIN
 -- PERFORM utl_file_utility.init();
   v_region := 'us-east-1';
   /* find tab details from file name */


   --DELETE FROM  ERROR_TABLE WHERE file_name= pi_filename;
  -- DELETE FROM  log_table WHERE file_name= pi_filename;


 BEGIN


   SELECT distinct table_name,table_name_stg INTO strict pi_ext_table,v_table_name_stg
   FROM  meta_EXTERNAL_TABLE
   WHERE upper(file_name) = upper(pi_filename);
 EXCEPTION
   WHEN NO_DATA_FOUND THEN
    raise notice 'error 1,%',sqlerrm;
    pi_ext_table := null;
    v_table_name_stg := null;
      RAISE USING errcode = 'NTFIP' ;
    when others then
        raise notice 'error others,%',sqlerrm;
 END;
```

```
 j :=1 ;

for rec in  cr2
 LOOP




  pi_ext_table     := rec.table_name;
  v_table_name_stg := rec.table_name_stg;
  v_col_list := null;


 IF pi_ext_table IS NOT NULL
  THEN
    --EXECUTE concat_ws('','truncate table  ' ,pi_ext_table) ;
   EXECUTE concat_ws('','truncate table  ' ,v_table_name_stg) ;




      SELECT distinct DELIMETER INTO STRICT v_delim
      FROM  meta_EXTERNAL_TABLE
      WHERE table_name = pi_ext_table;


      IF v_delim IS NOT NULL THEN
    SELECT distinct DELIMETER,
      FILE_NAME,
      DIRECTORY ,
      concat_ws('',' ',table_name_stg),
      case  header_exist when 'YES' then 'CSV HEADER' else 'CSV' end as header_exist
    INTO STRICT v_delim,v_file_name,v_directory,v_table_name_stg,v_header
    FROM  meta_EXTERNAL_TABLE
    WHERE table_name = pi_ext_table
      AND DELIMETER IS NOT NULL;


    IF    upper(v_delim) = 'CSV'
    THEN
      v_sql := concat_ws('','SELECT aws_s3.table_import_FROM_s3 ( ''',
      v_table_name_stg,''','''',
      ''DELIMITER '''','''' CSV HEADER QUOTE '''''''''', aws_commons.create_s3_uri
( ''',
```

```
      v_directory,''',''',v_file_name,''', ''',v_region,'''))');
      ELSE
      v_sql := concat_ws('','SELECT aws_s3.table_import_FROM_s3(''',
          v_table_name_stg, ''','''', ''DELIMITER AS ''''^''''','''','','
         aws_commons.create_s3_uri
          ( ''',v_directory, ''','''',
          v_file_name, ''',',
           '''',v_region,''')
         )');
         raise notice 'v_sql , %',v_sql;
      begin
       EXECUTE  v_sql;
      EXCEPTION
        WHEN OTHERS THEN
          raise notice 'error 1';
        RAISE USING errcode = 'S3IMP' ;
      END;


      select count(col_list) INTO v_col_cnt
      from  meta_EXTERNAL_TABLE where table_name = pi_ext_table;







       -- raise notice 'v_sql 2, %',concat_ws('','update ',v_table_name_stg, ' set
col1 = col1||''',v_delim,''''');


      execute concat_ws('','update ',v_table_name_stg, ' set col1 =
col1||''',v_delim,''''');




      i :=1;
      FOR rec in cr1
      loop
      v_sql1 := concat_ws('',v_sql1,'split_part(col1,''',v_delim,''',', i,')',' as
',rec.col_list,',');
      v_sql2 := concat_ws('',v_sql2,rec.col_list,',');
  --    v_sql3 := concat_ws('',v_sql3,'rec.',rec.col_list,'::',rec.data_type,',');
```

```
      case
        WHEN upper(rec.data_type) = 'NUMERIC'
        THEN v_sql3 := concat_ws('',v_sql3,' case WHEN
length(trim(split_part(col1,''',v_delim,''',', i,'))) =0
               THEN null
                ELSE
                coalesce((trim(split_part(col1,''',v_delim,''',',
i,')))::NUMERIC,0)::',rec.data_type,' END as ',rec.col_list,',') ;
        WHEN UPPER(rec.data_type) = 'TIMESTAMP WITHOUT TIME ZONE' AND rec.date_mask =
'YYYYMMDD'
        THEN v_sql3 := concat_ws('',v_sql3,' case WHEN
length(trim(split_part(col1,''',v_delim,''',', i,'))) =0
               THEN null
                ELSE
                to_date(coalesce((trim(split_part(col1,''',v_delim,''',',
i,'))),''99990101''),''YYYYMMDD'')::',rec.data_type,' END as ',rec.col_list,',');
        WHEN UPPER(rec.data_type) = 'TIMESTAMP WITHOUT TIME ZONE' AND rec.date_mask =
'MM/DD/YYYY hh24:mi:ss'
        THEN v_sql3 := concat_ws('',v_sql3,' case WHEN
length(trim(split_part(col1,''',v_delim,''',', i,'))) =0
               THEN null
                ELSE
                to_date(coalesce((trim(split_part(col1,''',v_delim,''',',
i,'))),''01/01/9999 0024:00:00''),''MM/DD/YYYY hh24:mi:ss'')::',rec.data_type,' END as
',rec.col_list,',');
         ELSE
       v_sql3 := concat_ws('',v_sql3,' case WHEN
length(trim(split_part(col1,''',v_delim,''',', i,'))) =0
               THEN null
                ELSE
                 coalesce((trim(split_part(col1,''',v_delim,''',',
i,'))),''''):',rec.data_type,' END as ',rec.col_list,',') ;
      END case;


      i :=i+1;
      end loop;


        -- raise notice 'v_sql 3, %',v_sql3;
```

```
       SELECT trim(trailing ' ' FROM v_sql1) INTO v_sql1;
       SELECT trim(trailing ',' FROM v_sql1) INTO v_sql1;


       SELECT trim(trailing ' ' FROM v_sql2) INTO v_sql2;
       SELECT trim(trailing ',' FROM v_sql2) INTO v_sql2;


       SELECT trim(trailing ' ' FROM v_sql3) INTO v_sql3;
       SELECT trim(trailing ',' FROM v_sql3) INTO v_sql3;


       END IF;
      raise notice 'v_delim , %',v_delim;


     EXECUTE concat_ws('','SELECT COUNT(*) FROM ',v_table_name_stg)  INTO v_cnt;


    raise notice 'stg cnt , %',v_cnt;


    /* if upper(v_delim) = 'CSV' then
       v_sql_ins := concat_ws('', ' SELECT * from ' ,v_table_name_stg );
     else
      -- v_sql_ins := concat_ws('',' SELECT ',v_sql1,'  from (select col1 from
 ' ,v_table_name_stg , ')sub ');
       v_sql_ins := concat_ws('',' SELECT ',v_sql3,'  from (select col1 from
 ' ,v_table_name_stg , ')sub ');
       END IF;*/


v_chunk := v_cnt/100;




for i in 1..101
loop
     BEGIN
    -- raise notice 'v_sql , %',v_sql;
       -- raise notice 'Chunk number , %',i;
       v_sql_ins := concat_ws('',' SELECT ',v_sql3,'  from (select col1 from
 ' ,v_table_name_stg , ' offset ',v_chunk*(i-1), ' limit ',v_chunk,') sub ');
```

```
    v_sql := concat_ws('','insert into  ', pi_ext_table ,' ', v_sql_ins);
    -- raise notice 'select statement , %',v_sql_ins;
         -- v_sql := null;
    -- EXECUTE concat_ws('','insert into  ', pi_ext_table ,' ', v_sql_ins, 'offset
',v_chunk*(i-1), ' limit ',v_chunk );
    --v_sql := concat_ws('','insert into  ', pi_ext_table ,' ', v_sql_ins );


    -- raise notice 'insert statement , %',v_sql;


   raise NOTICE 'CHUNK START %',v_chunk*(i-1);
  raise NOTICE 'CHUNK END %',v_chunk;


    EXECUTE v_sql;


 EXCEPTION
      WHEN OTHERS THEN
      -- v_sql_ins := concat_ws('',' SELECT ',v_sql1, '  from (select col1 from
' ,v_table_name_stg , ' )sub ');
        -- raise notice 'Chunk number for cursor , %',i;


   raise NOTICE 'Cursor - CHUNK START %',v_chunk*(i-1);
  raise NOTICE 'Cursor -  CHUNK END %',v_chunk;
        v_sql_ins := concat_ws('',' SELECT ',v_sql3, '  from (select col1 from
' ,v_table_name_stg , ' )sub ');


        v_final_sql := REPLACE (v_sql_ins, ''''::text, ''''''::text);
       -- raise notice 'v_final_sql %',v_final_sql;
        v_sql :=concat_ws('','do $a$ declare  r refcursor;v_sql text; i
numeric;v_conname text;  v_typ  ',pi_ext_table,'[]; v_rec  ','record',';
          begin
```

```
          open r for execute ''select col1 from ',v_table_name_stg ,'  offset
 ',v_chunk*(i-1), ' limit ',v_chunk,''';
          loop
          begin
          fetch r into v_rec;
          EXIT WHEN NOT FOUND;




          v_sql := concat_ws('''','insert into  ',pi_ext_table,' SELECT ',REPLACE
 (v_sql3, ''''::text, ''''''::text) , '  from ( select '''''',v_rec.col1,'''''' as
 col1) v'');
           execute v_sql;


          exception
           when others then
         v_sql := ''INSERT INTO  ERROR_TABLE VALUES (concat_ws(''''''''','''Error
 Name: ''',$$''||SQLERRM||''$$,''''Error State: ''',''''''||
SQLSTATE||'''''',''''record : ''',$$''||v_rec.col1||''$$),'''''||
pi_filename||'''''',now())'';


             execute v_sql;
            continue;
          end ;
          end loop;
          close r;
          exception
          when others then
        raise;
          end ; $a$');
     -- raise notice ' inside excp v_sql %',v_sql;
         execute v_sql;
     --  raise notice 'v_sql %',v_sql;
      END;
  END LOOP;
     ELSE


     SELECT distinct DELIMETER,FILE_NAME,DIRECTORY ,concat_ws('',' ',table_name_stg),
       case  header_exist when 'YES' then 'CSV HEADER' else 'CSV' end as header_exist
       INTO STRICT v_delim,v_file_name,v_directory,v_table_name_stg,v_header
```

```
    FROM  meta_EXTERNAL_TABLE
    WHERE table_name = pi_ext_table                  ;
    v_sql := concat_ws('','SELECT aws_s3.table_import_FROM_s3(''',
      v_table_name_stg, ''','''', ''DELIMITER AS ''''#'''' ',v_header,' '','','
     aws_commons.create_s3_uri
      ( ''',v_directory, ''','''',
      v_file_name, ''',',
       '''',v_region,''')
     )');
        EXECUTE  v_sql;


    FOR rec in cr1
    LOOP


     IF rec.start_pos IS NULL AND rec.END_pos IS NULL AND rec.no_position = 'recnum'
     THEN
       v_rec_val := 1;
     ELSE


      case
        WHEN upper(rec.data_type) = 'NUMERIC'
        THEN v_sql1 := concat_ws('',' case WHEN length(trim(substring(COL1,
',rec.start_pos ,',', rec.END_pos,'-',rec.start_pos ,'+1))) =0
               THEN null
                ELSE
                coalesce((trim(substring(COL1, ',rec.start_pos ,',',
rec.END_pos,'-',rec.start_pos ,'+1)))::NUMERIC,0)::',rec.data_type,' END as
',rec.col_list,',') ;
        WHEN UPPER(rec.data_type) = 'TIMESTAMP WITHOUT TIME ZONE' AND rec.date_mask =
'YYYYMMDD'
        THEN v_sql1 := concat_ws('','case WHEN length(trim(substring(COL1,
',rec.start_pos ,',', rec.END_pos,'-',rec.start_pos ,'+1))) =0
               THEN null
                ELSE
                to_date(coalesce((trim(substring(COL1, ',rec.start_pos ,',',
rec.END_pos,'-',rec.start_pos ,'+1))),''99990101''),''YYYYMMDD''):',rec.data_type,'
END as ',rec.col_list,',');
        WHEN UPPER(rec.data_type) = 'TIMESTAMP WITHOUT TIME ZONE' AND rec.date_mask =
'YYYYMMDDHH24MISS'
        THEN v_sql1 := concat_ws('','case WHEN length(trim(substring(COL1,
',rec.start_pos ,',', rec.END_pos,'-',rec.start_pos ,'+1))) =0
```

```
               THEN null
                ELSE
                to_date(coalesce((trim(substring(COL1, ',rec.start_pos ,',',
rec.END_pos,'-',rec.start_pos ,'+1))),''9999010100240000''),''YYYYMMDDHH24MISS''):',rec.data_
END as ',rec.col_list,',');
         ELSE
       v_sql1 := concat_ws('',' case WHEN length(trim(substring(COL1,
',rec.start_pos ,',', rec.END_pos,'-',rec.start_pos ,'+1))) =0
               THEN null
                ELSE
                 coalesce((trim(substring(COL1, ',rec.start_pos ,',',
rec.END_pos,'-',rec.start_pos ,'+1))),''''):',rec.data_type,' END as
',rec.col_list,',') ;
      END case;


     END IF;
     v_col_list := concat_ws('',v_col_list ,v_sql1);
    END LOOP;




          SELECT trim(trailing ' ' FROM v_col_list) INTO v_col_list;
          SELECT trim(trailing ',' FROM v_col_list) INTO v_col_list;


          v_sql_col   :=  concat_ws('',trim(trailing ',' FROM v_col_list) , ' FROM
',v_table_name_stg,' WHERE col1 IS NOT NULL AND length(col1)>0 ');




          v_sql_dynamic := v_sql_col;


          EXECUTE  concat_ws('','SELECT COUNT(*) FROM ',v_table_name_stg) INTO v_cnt;




        IF v_rec_val = 1 THEN
            v_sql_ins := concat_ws('',' select row_number() over(order by ctid) as
line_number ,' ,v_sql_dynamic) ;
```

```
        ELSE
              v_sql_ins := concat_ws('',' SELECT' ,v_sql_dynamic) ;
          END IF;


    BEGIN
      EXECUTE concat_ws('','insert into  ', pi_ext_table ,' ', v_sql_ins);
          EXCEPTION
             WHEN OTHERS THEN
         IF v_rec_val = 1 THEN
                v_final_sql := ' select row_number() over(order by ctid) as
line_number ,col1 from ';
               ELSE
                v_final_sql := ' SELECT col1 from';
              END IF;
      v_sql :=concat_ws('','do $a$ declare  r refcursor;v_rec_val numeric :=
',coalesce(v_rec_val,0),';line_number numeric; col1 text; v_typ  ',pi_ext_table,'[];
v_rec  ',pi_ext_table,';
            begin
            open r for execute ''',v_final_sql, ' ',v_table_name_stg,' WHERE col1 IS
NOT NULL AND length(col1)>0 '' ;
            loop
            begin
            if   v_rec_val = 1 then
            fetch r into line_number,col1;
            else
            fetch r into col1;
            end if;


            EXIT WHEN NOT FOUND;
             if v_rec_val = 1 then
             select line_number,',trim(trailing ',' FROM v_col_list) ,' into v_rec;
             else
               select ',trim(trailing ',' FROM v_col_list) ,' into v_rec;
             end if;


            insert into  ',pi_ext_table,' select v_rec.*;
             exception
             when others then
```

```
              INSERT INTO  ERROR_TABLE VALUES (concat_ws(''''',''Error Name:
'',SQLERRM,''Error State: '',SQLSTATE,''record : '',v_rec),''',pi_filename,''',now());
              continue;
             end ;
              end loop;
            close r;
             exception
             when others then
             raise;
             end ; $a$');
        execute v_sql;


    END;


        END IF;


  EXECUTE concat_ws('','SELECT COUNT(*) FROM  ' ,pi_ext_table)   INTO proc_rec_COUNT;


  EXECUTE concat_ws('','SELECT COUNT(*) FROM  error_table WHERE file_name
='''',pi_filename,''' and processed_time::date = clock_timestamp()::date')  INTO
error_rec_COUNT;


  EXECUTE concat_ws('','SELECT COUNT(*) FROM ',v_table_name_stg)   INTO tot_rec_COUNT;


  INSERT INTO  log_table values(pi_filename,now(),tot_rec_COUNT,proc_rec_COUNT,
error_rec_COUNT);


  raise notice 'v_directory, %',v_directory;


  raise notice 'pi_filename, %',pi_filename;


  raise notice 'v_region, %',v_region;
```

```
  perform aws_s3.query_export_to_s3('SELECT
 replace(trim(substring(error_details,position(''('' in
 error_details)+1),'')''),'','','';''),file_name,processed_time FROM  error_table WHERE
 file_name = '''||pi_filename||'''',
   aws_commons.create_s3_uri(v_directory, pi_filename||'.bad', v_region),
   options :='FORmat csv, header, delimiter $$,$$'
   );


raise notice 'v_directory, %',v_directory;


   raise notice 'pi_filename, %',pi_filename;


   raise notice 'v_region, %',v_region;


  perform aws_s3.query_export_to_s3('SELECT * FROM  log_table WHERE file_name = '''||
pi_filename||'''',
   aws_commons.create_s3_uri(v_directory, pi_filename||'.log', v_region),
   options :='FORmat csv, header, delimiter $$,$$'
   );




   END IF;
 j := j+1;
 END LOOP;


       RETURN 'OK';
EXCEPTION
    WHEN  OTHERS THEN
  raise notice 'error %',sqlerrm;
   ERCODE=SQLSTATE;
   IF ERCODE = 'NTFIP' THEN
     v_sqlerrm := concat_Ws('',sqlerrm,'No data for the filename');
   ELSIF ERCODE = 'S3IMP' THEN
    v_sqlerrm := concat_Ws('',sqlerrm,'Error While exporting the file from S3');
   ELSE
      v_sqlerrm := sqlerrm;
   END IF;
```

```
 select distinct directory into v_directory from  meta_EXTERNAL_TABLE;



 raise notice 'exc v_directory, %',v_directory;


   raise notice 'exc pi_filename, %',pi_filename;


   raise notice 'exc v_region, %',v_region;


  perform aws_s3.query_export_to_s3('SELECT * FROM  error_table WHERE file_name = '''||
pi_filename||'''',
   aws_commons.create_s3_uri(v_directory, pi_filename||'.bad', v_region),
   options :='FORmat csv, header, delimiter $$,$$'
   );
    RETURN null;
END;
$function$
```

Migrazione di indici basati su funzioni da Oracle a PostgreSQL

Veeranjaneyulu Grandhi e Navakanth Talluri, Amazon Web Services

Riepilogo

Gli indici sono un modo comune per migliorare le prestazioni del database. Un indice consente al server del database di trovare e recuperare righe specifiche molto più velocemente di quanto potrebbe fare senza un indice. Ma gli indici aggiungono anche un sovraccarico all'intero sistema di database, quindi devono essere usati in modo sensato. Gli indici basati su funzioni, che si basano su una funzione o un'espressione, possono includere più colonne ed espressioni matematiche. Un indice basato su funzioni migliora le prestazioni delle query che utilizzano l'espressione dell'indice.

A livello nativo, PostgreSQL non supporta la creazione di indici basati su funzioni utilizzando funzioni la cui volatilità è definita stabile. Tuttavia, è possibile creare funzioni simili con volatilità e utilizzarle nella creazione di indici. `IMMUTABLE`

Una `IMMUTABLE` funzione non può modificare il database ed è garantito che restituirà gli stessi risultati con gli stessi argomenti per sempre. Questa categoria consente all'ottimizzatore di valutare preventivamente la funzione quando una query la richiama con argomenti costanti.

Questo modello aiuta a migrare gli indici basati sulle funzioni Oracle quando vengono utilizzati con funzioni come `to_char`o `to_date`, e verso `to_number` l'equivalente PostgreSQL.

Prerequisiti e limitazioni

Prerequisiti

- Un account Amazon Web Services (AWS) attivo
- Un'istanza di database Oracle di origine con il servizio listener configurato e funzionante
- Familiarità con i database PostgreSQL

Limitazioni

- Il limite di dimensione del database è di 64 TB.
- Le funzioni utilizzate nella creazione dell'indice devono essere IMMUTABILI.

Versioni del prodotto

- Tutte le edizioni del database Oracle per le versioni 11g (versioni 11.2.0.3.v1 e successive) e fino a 12.2 e 18c
- PostgreSQL 9.6 e versioni successive

Architecture

Stack tecnologico di origine

- Un database Oracle in locale o su un'istanza Amazon Elastic Compute Cloud (Amazon EC2) o un'istanza Amazon RDS for Oracle DB

Stack tecnologico Target

- Qualsiasi motore PostgreSQL

Tools (Strumenti)

- pGAdmin 4 è uno strumento di gestione open source per Postgres. Lo strumento pgAdmin 4 fornisce un'interfaccia grafica per la creazione, la manutenzione e l'utilizzo di oggetti di database.
- Oracle SQL Developer è un ambiente di sviluppo integrato (IDE) per lo sviluppo e la gestione di database Oracle in implementazioni tradizionali e cloud.

Epiche

Crea un indice basato su funzioni utilizzando una funzione predefinita

Operazione	Description	Competenze richieste

Crea un indice basato su funzioni su una colonna utilizzando la funzione to_char.	Utilizzate il codice seguente per creare l'indice basato sulle funzioni.
	`postgres=# create table funcindex( col1 timestamp without time zone);`
	`CREATE TABLE`
		DBA, sviluppatore di app

Operazione	Description	Competenze richieste
	`postgres=# insert into funcindex values (now()); INSERT 0 1 postgres=# select * from funcindex; col1 ---------- 2022-08-09 16:00:57.77414 (1 rows) postgres=# create index funcindex_idx on funcindex(to_char(col1,'DD-MM-YYYY HH24:MI:SS')); ERROR: functions in index expression must be marked IMMUTABLE` **Note** PostgreSQL non consente la creazione di un indice basato su funzioni senza la clausola. `IMMUTABLE`	
Verifica la volatilità della funzione.	Per controllare la volatilità della funzione, usa il codice nella sezione Informazioni aggiuntive.	DBA

Crea indici basati su funzioni utilizzando una funzione wrapper

Operazione	Description	Competenze richieste
Crea una funzione wrapper.	Per creare una funzione wrapper, usa il codice nella sezione Informazioni aggiuntive.	Sviluppatore PostgreSQL
Crea un indice utilizzando la funzione wrapper.	Utilizzate il codice nella sezione Informazioni aggiuntive per creare una funzione definita dall'utente con la parola chiave `IMMUTABLE` nello stesso schema dell'applicazione e fate riferimento ad essa nello script di creazione dell'indice. Se una funzione definita dall'utente viene creata in uno schema comune (dall'esempio precedente), aggiornatela come illustrato. `search_path` `ALTER ROLE <ROLENAME> set search_path=$user, COMMON;`	DBA, sviluppatore PostgreSQL

Convalida la creazione dell'indice

Operazione	Description	Competenze richieste
Convalida la creazione dell'indice.	Verifica che l'indice debba essere creato, in base ai modelli di accesso alle query.	DBA
Verifica che l'indice possa essere usato.	Per verificare se l'indice basato sulla funzione viene rilevato da PostgreSQL Optimizer, esegui un'istruzione SQL utilizzando explain o explain analyze. Usa il codice nella sezione Informazioni aggiuntive. Se possibile, raccogli anche le statistiche della tabella. **Note** Se notate il piano di spiegazione, l'ottimizzatore PostgreSQL ha scelto un indice basato sulle funzioni a causa della condizione del predicato.	DBA

Risorse correlate

- Indici basati sulle funzioni (documentazione Oracle)
- Indici sulle espressioni (documentazione PostgreSQL)
- Volatilità di PostgreSQL (documentazione PostgreSQL)
- PostgreSQL search_path (documentazione PostgreSQL)

- Playbook sulla migrazione da Oracle Database 19c ad Amazon Aurora PostgreSQL

Informazioni aggiuntive

Crea una funzione wrapper

```
CREATE OR REPLACE FUNCTION myschema.to_char(var1 timestamp without time zone, var2
 varchar) RETURNS varchar AS $BODY$ select to_char(var1, 'YYYYMMDD'); $BODY$ LANGUAGE
 sql IMMUTABLE;
```

Crea un indice utilizzando la funzione wrapper

```
postgres=# create function common.to_char(var1 timestamp without time zone, var2
 varchar) RETURNS varchar AS $BODY$ select to_char(var1, 'YYYYMMDD'); $BODY$ LANGUAGE
 sql IMMUTABLE;
CREATE FUNCTION
postgres=# create index funcindex_idx on funcindex(common.to_char(col1,'DD-MM-YYYY
 HH24:MI:SS'));
CREATE INDEX
```

Controlla la volatilità della funzione

```
SELECT DISTINCT p.proname as "Name",p.provolatile as "volatility" FROM
 pg_catalog.pg_proc p
 LEFT JOIN pg_catalog.pg_namespace n ON n.oid = p.pronamespace
 LEFT JOIN pg_catalog.pg_language l ON l.oid = p.prolang
 WHERE n.nspname OPERATOR(pg_catalog.~) '^(pg_catalog)$' COLLATE pg_catalog.default AND
 p.proname='to_char'GROUP BY p.proname,p.provolatile
ORDER BY 1;
```

Verifica che l'indice possa essere utilizzato

```
explain analyze <SQL>


postgres=# explain select col1 from funcindex where common.to_char(col1,'DD-MM-YYYY
 HH24:MI:SS') = '09-08-2022 16:00:57';
                                                    QUERY PLAN
-------------------------------------------------------------------------------------------
 Index Scan using funcindex_idx on funcindex  (cost=0.42..8.44 rows=1 width=8)
```

```
   Index Cond: ((common.to_char(col1, 'DD-MM-YYYY HH24:MI:SS'::character
 varying))::text = '09-08-2022 16:00:57'::text)
(2 rows)
```

Migrazione delle funzioni native di Oracle su PostgreSQL utilizzando le estensioni

Pinesh Singal, Amazon Web Services

Riepilogo

Questo modello di migrazione fornisce step-by-step indicazioni per la migrazione di un'istanza di database Amazon Relational Database Service (Amazon RDS) per Oracle verso un database Amazon RDS for PostgreSQL o Amazon Aurora PostgreSQL Edition modificando le estensioni and nel codice integrato nativo di PostgreSQL (). `aws_oracle_ext` `orafce` `psql` Ciò consentirà di risparmiare tempo di elaborazione.

Il modello descrive una strategia di migrazione manuale offline senza tempi di inattività per un database di origine Oracle da più terabyte con un numero elevato di transazioni.

Il processo di migrazione utilizza AWS Schema Conversion Tool (AWS SCT) con le `orafce` estensioni `aws_oracle_ext` e per convertire uno schema di database Amazon RDS for Oracle in uno schema di database compatibile con Amazon RDS for PostgreSQL o Aurora PostgreSQL. Quindi il codice viene modificato manualmente in codice integrato nativo supportato da PostgreSQL. `psql` Questo perché le chiamate di estensione influiscono sull'elaborazione del codice sul server di database PostgreSQL e non tutto il codice di estensione è completamente conforme o compatibile con il codice PostgreSQL.

Questo modello si concentra principalmente sulla migrazione manuale dei codici SQL utilizzando AWS SCT e le estensioni `aws_oracle_ext` e. `orafce` Le estensioni già utilizzate vengono convertite in funzionalità integrate native di `psql` PostgreSQL (). Quindi rimuovete tutti i riferimenti alle estensioni e convertite i codici di conseguenza.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Sistema operativo (Windows o Mac) o EC2 istanza Amazon (attiva e funzionante)
- Oracle

Limitazioni

Non tutte le funzioni Oracle che utilizzano `aws_oracle_ext` o `orafce` le estensioni possono essere convertite in funzioni PostgreSQL native. Potrebbe essere necessario rielaborarlo manualmente per compilarlo con le librerie PostgreSQL.

Uno svantaggio dell'utilizzo delle estensioni AWS SCT è la lentezza delle prestazioni nell'esecuzione e nel recupero dei risultati. Il suo costo può essere compreso dal semplice piano EXPLAIN di PostgreSQL (piano di esecuzione di una dichiarazione) sulla migrazione delle funzioni Oracle alla *SYSDATE* funzione *NOW()* PostgreSQL tra tutti e tre i codici *aws_oracle_ext* (*orafce*, *psql* e predefinito), come spiegato nella sezione Controllo del confronto delle prestazioni del documento allegato.

Versioni del prodotto

- Fonte: database Amazon RDS for Oracle 10.2 e versioni successive (per 10.x), 11g (11.2.0.3.v1 e versioni successive) e fino a 12.2, 18c e 19c (e versioni successive) per Enterprise Edition, Standard Edition, Standard Edition 1 e Standard Edition 2
- Target: database compatibile con Amazon RDS for PostgreSQL o Aurora PostgreSQL 9.4 e versioni successive (per 9.x), 10.x, 11.x, 12.x, 13.x e 14.x (e versioni successive)
- AWS SCT: versione più recente (questo modello è stato testato con 1.0.632)
- Oracle: ultima versione (questo pattern è stato testato con 3.9.0)

Architecture

Stack di tecnologia di origine

- Un'istanza di database Amazon RDS for Oracle con la versione 12.1.0.2.v18

Stack tecnologico Target

- Un'istanza di database compatibile con Amazon RDS for PostgreSQL o Aurora PostgreSQL con versione 11.5

Architettura di migrazione del database

Il diagramma seguente rappresenta l'architettura di migrazione del database tra i database Oracle di origine e PostgreSQL di destinazione. L'architettura include AWS Cloud, un cloud privato virtuale (VPC), zone di disponibilità, una sottorete privata, un database Amazon RDS per Oracle, AWS

SCT, Amazon RDS per PostgreSQL o Aurora PostgreSQL, estensioni per Oracle (and) e file SQL (Structured Query Language). `aws_oracle_ext` `orafce`

1. Avvia l'istanza DB di Amazon RDS for Oracle (database di origine).
2. Usa AWS SCT con `aws_oracle_ext` i pacchetti di `orafce` estensione per convertire il codice sorgente da Oracle a PostreSQL.
3. La conversione produce file.sql migrati supportati da PostgreSQL.
4. Converti manualmente i codici di estensione Oracle non convertiti in codici `psql` PostgreSQL ().
5. La conversione manuale produce file.sql convertiti supportati da PostgreSQL.
6. Esegui questi file.sql sulla tua istanza database Amazon RDS for PostgreSQL (DB di destinazione).

Tools (Strumenti)

Strumenti

Servizi AWS

- AWS SCT - AWS Schema Conversion Tool (AWS SCT) converte lo schema di database esistente da un motore di database a un altro. Puoi convertire lo schema relazionale OLTP (Online Transactional Processing) o lo schema di data warehouse. Lo schema convertito è adatto per un'istanza DB Amazon RDS for MySQL, un cluster DB Amazon Aurora, un'istanza DB Amazon RDS for PostgreSQL o un cluster Amazon Redshift. Lo schema convertito può essere utilizzato anche con un database su un' EC2 istanza Amazon o archiviato come dati in un bucket Amazon S3.

  AWS SCT fornisce un'interfaccia utente basata su progetti per convertire automaticamente lo schema del database di origine in un formato compatibile con l'istanza Amazon RDS di destinazione.

  Puoi utilizzare AWS SCT per eseguire la migrazione da un database di origine Oracle a uno qualsiasi degli obiettivi elencati in precedenza. Utilizzando AWS SCT, puoi esportare le definizioni degli oggetti del database di origine come schema, viste, stored procedure e funzioni.

  Puoi usare AWS SCT per convertire i dati da Oracle ad Amazon RDS for PostgreSQL o Amazon Aurora PostgreSQL Compatible Edition.

In questo modello, usi AWS SCT per convertire e migrare il codice Oracle in PostgreSQL utilizzando le estensioni `aws_oracle_ext` e `orafce` migrando manualmente i codici di estensione in codice integrato predefinito o nativo. `psql`

- Il pacchetto di estensione AWS SCT è un modulo aggiuntivo che emula le funzioni presenti nel database di origine necessarie per la conversione degli oggetti nel database di destinazione. Prima di poter installare il pacchetto di estensione AWS SCT, devi convertire lo schema del database.

  Quando converti lo schema del database o del data warehouse, AWS SCT aggiunge uno schema aggiuntivo al database di destinazione. Questo schema implementa le funzioni di sistema SQL del database di origine richieste durante la scrittura dello schema convertito nel database di destinazione. Lo schema aggiuntivo viene chiamato schema del pacchetto di estensione.

  Lo schema del pacchetto di estensione per i database OLTP è denominato in base al database di origine. Per i database Oracle, lo schema del pacchetto di estensione è`AWS_ORACLE_EXT`.

Altri strumenti

- Oracle: Orafce è un modulo che implementa funzioni, tipi di dati e pacchetti compatibili con Oracle. È uno strumento open source con una licenza Berkeley Source Distribution (BSD) in modo che chiunque possa utilizzarlo. Il `orafce` modulo è utile per la migrazione da Oracle a PostgreSQL perché ha molte funzioni Oracle implementate in PostgreSQL.

Codice

Per un elenco di tutto il codice comunemente usato e migrato da Oracle a PostgreSQL per evitare l'uso del codice di estensione AWS SCT, consulta il documento allegato.

Epiche

Configurazione del database di origine Amazon RDS for Oracle

Operazione	Description	Competenze richieste
Creare l'istanza del database Oracle.	Crea un'istanza di database compatibile con Amazon RDS for Oracle o Aurora PostgreSQL dalla console Amazon RDS.	Informazioni generali su AWS, DBA

Operazione	Description	Competenze richieste
Configura i gruppi di sicurezza .	Configura i gruppi di sicurezza in entrata e in uscita.	Informazioni generali su AWS
Crea il database.	Crea il database Oracle con gli utenti e gli schemi necessari.	Informazioni generali su AWS, DBA
Crea gli oggetti.	Crea oggetti e inserisci dati nello schema.	DBA

Configurazione del database di destinazione Amazon RDS for PostgreSQL

Operazione	Description	Competenze richieste
Crea l'istanza del database PostgreSQL.	Crea un'istanza di database Amazon RDS for PostgreSQL o Amazon Aurora PostgreSQL dalla console Amazon RDS.	Informazioni generali su AWS, DBA
Configura i gruppi di sicurezza .	Configura i gruppi di sicurezza in entrata e in uscita.	Informazioni generali su AWS
Crea il database.	Crea il database PostgreSQ L con gli utenti e gli schemi necessari.	Informazioni generali su AWS, DBA
Convalida le estensioni.	Assicurati che `aws_oracl e_ext` e `orafce` siano installati e configurati correttamente nel database PostgreSQL.	DBA
Verifica che il database PostgreSQL sia disponibile.	Assicurati che il database PostgreSQL sia attivo e funzionante.	DBA

Migra lo schema Oracle in PostgreSQL utilizzando AWS SCT e le estensioni

Operazione	Description	Competenze richieste
Installa AWS SCT.	Installa la versione più recente di AWS SCT.	DBA
Configura AWS SCT.	Configura AWS SCT con i driver Java Database Connectivity (JDBC) per Oracle () `ojdbc8.jar` e PostgreSQL (). `postgresql-42.2.5.jar`	DBA
Abilita il pacchetto di estensione o il modello di estensione AWS SCT.	In AWS SCT Project Settings, abilita l'implementazione di funzioni integrate con `aws_oracle_ext` ed `orafce` estensioni per lo schema del database Oracle.	DBA
Convertire lo schema.	In AWS SCT, scegli Converti schema per convertire lo schema da Oracle a PostgreSQL e generare i file.sql.	DBA

Converti il codice di estensione AWS SCT in codice psql

Operazione	Description	Competenze richieste
Convertire manualmente il codice.	Converti manualmente ogni riga di codice supportato dall'estensione in codice integrato `psql` predefini to, come dettagliato nel documento allegato. Ad	DBA

Operazione	Description	Competenze richieste
	esempio, change `AWS_ORACLE_EXT.SYSDATE()` o `ORACLE.SYSDATE()` to. `NOW()`	
Convalida il codice	(Facoltativo) Convalida ogni riga di codice eseguendo la temporaneamente nel database PostgreSQL.	DBA
Crea oggetti nel database PostgreSQL.	Per creare oggetti nel database PostgreSQL, esegui i file.sql generati da AWS SCT e modificati nei due passaggi precedenti.	DBA

Risorse correlate

- Database
  - Oracle su Amazon RDS
  - PostgreSQL su Amazon RDS
  - Lavorare con Amazon Aurora PostgreSQL
  - Piano PostgreSQL EXPLAIN
- AWS SCT
  - Panoramica dello Schema Conversion Tool di AWS
  - Guida per l'utente di AWS SCT
  - Utilizzo dell'interfaccia utente AWS SCT
  - Utilizzo di Oracle Database come sorgente per AWS SCT
- Estensioni per AWS SCT
  - Utilizzo del pacchetto di estensione AWS SCT
  - Funzionalità Oracle (en)
  - Oracle PGXN

- GitHub orafce

Informazioni aggiuntive

Per ulteriori informazioni, segui i comandi dettagliati, con sintassi ed esempi, per convertire manualmente il codice nel documento allegato.

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Esegui la migrazione di un database Db2 da Amazon a EC2 Aurora compatibile con MySQL utilizzando AWS DMS

Pinesh Singal, Amazon Web Services

Riepilogo

Dopo aver migrato il database IBM Db2 for LUW su Amazon Elastic Compute Cloud (EC2Amazon), prendi in considerazione la possibilità di riprogettare il database passando a un database nativo per il cloud di Amazon Web Services (AWS). Questo modello riguarda la migrazione di un database IBM Db2 for LUW in esecuzione su un' EC2 istanza Amazon verso un database Edition compatibile con Amazon Aurora MySQL su AWS.

Il modello descrive una strategia di migrazione online con tempi di inattività minimi per un database di origine Db2 da più terabyte con un numero elevato di transazioni.

Questo modello utilizza AWS Schema Conversion Tool (AWS SCT) per convertire lo schema del database Db2 in uno schema compatibile con Aurora MySQL. Quindi il modello utilizza AWS Database Migration Service (AWS DMS) per migrare i dati dal database Db2 al database Aurora compatibile con MySQL. Saranno necessarie conversioni manuali per il codice che non viene convertito da AWS SCT.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo con un cloud privato virtuale (VPC)
- AWS SCT
- AWS DMS

Versioni del prodotto

- Versione più recente di AWS SCT
- Db2 per Linux versione 11.1.4.4 e successive

Architecture

Stack tecnologico di origine

- DB2/Linux x86-64 bit montato su un'istanza EC2

Stack tecnologico Target

- Un'istanza di database Edition compatibile con Amazon Aurora MySQL

Architettura di origine e destinazione

Il diagramma seguente mostra l'architettura di migrazione dei dati tra i database Aurora compatibili con MySQL di origine e Db2 di destinazione. L'architettura sul cloud AWS include un cloud privato virtuale (VPC) (Virtual Private Cloud), una zona di disponibilità, una sottorete pubblica per l'istanza Db2 e l'istanza di replica AWS DMS e una sottorete privata per il database Aurora compatibile con MySQL.

Tools (Strumenti)

Servizi AWS

- Amazon Aurora è un motore di database relazionale completamente gestito creato per il cloud e compatibile con MySQL e PostgreSQL.
- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- AWS Schema Conversion Tool (AWS SCT) supporta migrazioni di database eterogenei convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione. AWS SCT supporta come sorgente IBM Db2 per le versioni LUW 9.1, 9.5, 9.7, 10.1, 10.5, 11.1 e 11.5.

Best practice

Per le best practice, consulta Best practice for AWS Database Migration Service.

Epiche

Configura il database IBM Db2 di origine

Operazione	Description	Competenze richieste
Crea il database IBM Db2 su Amazon. EC2	Puoi creare un database IBM Db2 su un' EC2 istanza utilizzando un'Amazon Machine Image (AMI) da AWS Marketplace o installando il software Db2 su un'istanza. EC2 Avvia un' EC2 istanza selezionando un AMI per IBM Db2 (ad esempio, IBM Db2 v11.5.7 RHEL 7.9), che è simile a un database locale.	DBA, AWS generale
Configura i gruppi di sicurezza .	Configura le regole in entrata del gruppo di sicurezza VPC per SSH (Secure Shell) e TCP con le porte 22 e 50000, rispettivamente.	Informazioni generali su AWS
Crea l'istanza del database.	Crea una nuova istanza (utente) e un database (schema) oppure usa l'`db2inst1`istanza e il database di esempio predefini ti. 1. Connettiti all' EC2 istanza utilizzando il terminale per connetterti al database Db2. In alternativa, puoi installar e qualsiasi software client	DBA

Operazione	Description	Competenze richieste
	DB che si connetterà al database Db2. 2. Per impostare la password dell'utente db2inst1, esegui il comando. `sudo passwd db2inst1` 3. Per connetterti all'istanza db2inst1, esegui il comando. `sudo su - db2inst1` 4. Per connetterti al database Db2, esegui il comando. `db2` 5. Per connetterti al database di esempio, usa il comando`connect to sample`. In alternativa, connettiti al database che hai creato. 6. Dopo esserti connesso all'istanza del database, crea oggetti e inserisci dati in questi oggetti utilizzando le istruzioni SQL Db2.	
Verifica che l'istanza DB Db2 sia disponibile.	Per confermare che l'istanza del database Db2 è attiva e in esecuzione, usa il `Db2pd -` comando.	DBA

Configurare il database Aurora di destinazione compatibile con MySQL

Operazione	Description	Competenze richieste
Crea il database Aurora compatibile con MySQL.	Crea un database di compatibilità Amazon Aurora con MySQL dal servizio AWS RDS • Crea un database su Amazon Aurora con compatibilità MySQL e versione a tua scelta, ad esempio Aurora (MySQL) — 5.6.10a • Installa l'applicazione MySQL Workbench o il tuo software client DB preferito che ti consente di connetterti al database MySQL	DBA, AWS generale
Configura i gruppi di sicurezza .	Configura le regole in entrata del gruppo di sicurezza VPC per le connessioni SSH e TCP.	Informazioni generali su AWS
Verifica che il database Aurora sia disponibile.	Per assicurarti che il database Aurora compatibile con MySQL sia attivo e funzionante, procedi come segue: 1. Connect all' EC2 istanza tramite SSH. 2. Configura e connettiti all'istanza Aurora compatibile con MySQL da MySQL Workbench. Usa l'endpoint come nome host, come	DBA

Operazione	Description	Competenze richieste
	mostrato nell'esempio seguente. `mysql-cluster-instance-1.cokmvis0v46q.us-east-1.rds.amazonaws.com` 3. Crea e connettiti al nuovo schema (ad esempio,`mysql-sample-db2`). 4. Esegui le istruzioni MySQL per controllare gli schemi e gli oggetti nel database.	

Configurazione ed esecuzione di AWS SCT

Operazione	Description	Competenze richieste
Installa AWS SCT.	Scarica e installa l'ultima versione di AWS SCT (l'ultima versione corrente 1.0.628).	Informazioni generali su AWS
Configura AWS SCT.	1. Scarica i driver Java Database Connectivity (JDBC) per IBM Db2 (versione 4.22.X) e MySQL (8.x). 2. Per configurare i driver in AWS SCT, scegli Impostazioni, Impostazioni globali, Driver.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Crea un progetto AWS SCT.	Crea un progetto e un report AWS SCT che utilizzi Db2 per LUW come motore DB di origine e compatibile con Aurora MySQL per il motore DB di destinazione. Per identificare i privilegi necessari per connettersi a un database Db2 for LUW, consulta Usare Db2 LUW come sorgente per AWS SCT.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Convalida gli oggetti.	Scegli Carica schema, convalida gli oggetti. Aggiorna eventuali oggetti errati sul database di destinazione: 1. Connettiti al server compatibile con Amazon Aurora MySQL fornendo i dettagli della connessione e scegli Test di connessione. Sia le connessioni di origine che quelle di destinazione devono avere successo prima che AWS SCT possa avviare il report di migrazione. 2. Una volta completato il report, inserisci lo schema da convertire e scegli Fine. AWS SCT elenca tutti gli oggetti di origine e destinazione che vengono convertiti e presentano errori. 3. Rivedi gli errori e cancellali manualmente. 4. Dopo aver eliminato tutti gli errori, apri il menu contestuale (fai clic con il pulsante destro del mouse) per lo schema e scegli Carica schema. 5. Scegli Applica al database.	DBA, AWS generale

Operazione	Description	Competenze richieste
	6. In MySQL Workbench , connettiti al database Aurora compatibile con MySQL e controlla lo schema e gli oggetti.	

Configurazione ed esecuzione di AWS DMS

Operazione	Description	Competenze richieste
Creare un'istanza di replica.	Accedi alla Console di gestione AWS, accedi al servizio AWS DMS e crea un'istanza di replica con impostazioni valide per il gruppo di sicurezza VPC che hai configurato per i database di origine e di destinazione.	Informazioni generali su AWS
Crea endpoint.	Crea l'endpoint di origine per il database Db2 e crea l'endpoint di destinazione per il database Aurora compatibile con MySQL: 1. Crea un endpoint per IBM Db2 come origine scegliendo Seleziona istanza DB RDS e quindi scegliendo l'istanza Db2 che hai creato. I dettagli di configurazione dell'endp oint verranno compilati automaticamente.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	2. Nelle impostazioni specifiche dell'endpoint, aggiungi i seguenti attributi di connessione aggiuntivi. `CurrentLSN=<scan>; MaxKBytesPerRead=64; SetDataCaptureChanges=true` Se non menzioni questi attributi, la connessione di test dell'endpoint di origine non avrà esito positivo. Per ulteriori informazioni, consulta Using IBM Db2 LUW come fonte per AWS DMS. 3. Crea un endpoint compatibile con Aurora MySQL come destinazione scegliendo Seleziona istanza DB RDS e quindi scegliendo l'istanza compatibile con Aurora MySQL che hai creato. I dettagli di configurazione dell'endpoint verranno compilati automaticamente. Per ulteriori informazioni, consulta Usare un database compatibile con MySQL come destinazione per AWS Database Migration Service.	

Operazione	Description	Competenze richieste
	4. Testa gli endpoint di origine e di destinazione. Verifica che entrambi abbiano esito positivo e siano disponibili 5. Se il test fallisce, assicurati che le regole in entrata del gruppo di sicurezza siano valide.	

Operazione	Description	Competenze richieste
Crea attività di migrazione.	Crea una singola attività di migrazione o più attività di migrazione per il pieno carico e la convalida CDC o dei dati: 1. Per creare un'attività di migrazione del database, scegli l'istanza di replica, l'endpoint del database di origine, l'endpoint del database di destinazione. Specificate il tipo di migrazione come Migrazione dei dati esistenti (pieno carico), Replica solo le modifiche ai dati (CDC) o Migra i dati esistenti e replica le modifiche in corso (pieno carico e CDC). 2. In Table mapping, puoi configurare le regole di selezione e le regole di trasformazione nei formati GUI o JSON. 3. In Regole di selezione, seleziona lo schema, inserisci il nome della tabella e seleziona Azione (Includi/Escludi) da configurare (ad esempio, Schema: SAMPLE; Nome tabella:%, Azione: Include). 4. In Regole di trasformazione, seleziona l'obietti	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	vo (schema, tabella o colonna). Seleziona il nome dello schema e scegli l'azione (maiuscol e, prefisso, suffisso) ; ad esempio, Target: Schema`mysql-sample-db`; Azione: Crea lettere minuscole. 5. Attiva il monitoraggio di Amazon CloudWatch Logs.	
Pianifica il ciclo di produzione.	Conferma i tempi di inattività con le parti interessate, come i proprietari delle applicazioni, per eseguire AWS DMS nei sistemi di produzione.	Responsabile della migrazione
Esegui le attività di migrazion e.	1. Avvia l'attività AWS DMS con stato Pronto. 2. Monitora i log delle attività di migrazione in Amazon CloudWatch Logs per eventuali errori.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Convalida i dati.	Esamina i risultati e i dati delle attività di migrazione nei database Db2 di origine e MySQL di destinazione: 1. Se lo stato è Carica, completa la replica in corso, il caricamento completo con la migrazione dei dati CDC è completato e la convalida è in corso. 2. Connect al database Aurora compatibile con MySQL e controlla i dati. 3. Controlla le modifiche in corso inserendo o aggiornando i dati nel database Db2.	DBA
Interrompi le attività di migrazione.	Una volta completata correttamente la convalida dei dati, interrompi le attività di migrazione di convalida.	Informazioni generali su AWS

risoluzione dei problemi

Problema	Soluzione
Le connessioni di test di origine e destinazione di AWS SCT non funzionano.	Configura le versioni dei driver JDBC e le regole in entrata del gruppo di sicurezza VPC per accettare il traffico in entrata.
L'esecuzione del test dell'endpoint sorgente Db2 non riesce.	Configura l'impostazione di connessione aggiuntiva. `CurrentLSN=<scan>;`

Problema	Soluzione

L' AWSDMS operazione non riesce a connetter si alla sorgente Db2 e viene restituito il seguente errore.
 	
`database is recoverable if either or both of the database configura tion parameters LOGARCHMETH1 and LOGARCHMETH2 are set to ON`
	Per evitare l'errore, esegui i seguenti comandi:
	
	1. `$ db2 update db cfg for sample using LOGARCHMETH1 DISK:/home/ db2inst1/logs`
	2. `$ db2stop`
	3. `$ db2start`
	4. `$ db2 connect to sample`
	
	`SQL1116N A connection to or activation of database "SAMPLE" cannot be made because of BACKUP PENDING. SQLSTATE=57019`
	
	5. `$ db2 backup database sample to ../logs`
	
	`SQL2036N The path for the file or device "../logs" is not valid`
	
	6. `$ cd`
	7. `$ pwd`
	
	`/home/db2inst1`
	
	8. `$ mkdir /tmp/backup`
	9. `$ db2 backup database sample to / tmp/backup`
	
	`Backup successful. The timestamp for this backup image is : 201905300 84921`
	
	10. `$ db2 connect to sample`
	
	`Database Connection Information`

Problema	Soluzione

	`Database server = DB2/LINUX 9.7.1`
	`SQL authorization ID = DB2INST1`
	`Local database alias = SAMPLE`

Risorse correlate

Amazon EC2

- Amazon EC2
- Guide EC2 per gli utenti di Amazon

Database

- Database IBM Db2
- Amazon Aurora
- Lavorare con Amazon Aurora MySQL

AWS SCT

- Conversione dello schema AWS DMS
- Guida per l'utente di AWS Schema Conversion Tool
- Utilizzo dell'interfaccia utente AWS SCT
- Utilizzo di IBM Db2 LUW come sorgente per AWS SCT

AWS DMS

- AWS Database Migration Service
- Guida per l'utente di AWS Database Migration Service
- Fonti per la migrazione dei dati
- Obiettivi per la migrazione dei dati
- AWS Database Migration Service e AWS Schema Conversion Tool ora supportano IBM Db2 LUW come sorgente (post di blog)

- Migrazione di applicazioni che eseguono database relazionali su AWS

Esegui la migrazione di un database Microsoft SQL Server da Amazon EC2 ad Amazon DocumentDB utilizzando AWS DMS

Umamaheswara Nooka, Amazon Web Services

Riepilogo

Questo modello descrive come utilizzare AWS Database Migration Service (AWS DMS) per migrare un database Microsoft SQL Server ospitato su un'istanza Amazon Elastic Compute Cloud (Amazon EC2) verso un database Amazon DocumentDB (con compatibilità MongoDB).

Il task di replica AWS DMS legge la struttura delle tabelle del database SQL Server, crea la raccolta corrispondente in Amazon DocumentDB ed esegue una migrazione a pieno carico.

Puoi anche utilizzare questo modello per migrare un'istanza DB SQL Server o Amazon Relational Database Service (Amazon RDS) per SQL Server locale su Amazon DocumentDB. Per ulteriori informazioni, consulta la guida Migrazione dei database Microsoft SQL Server al cloud AWS sul sito Web AWS Prescriptive Guidance.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un database SQL Server esistente su un'istanza. EC2
- Ruolo fisso del database (db_owner) assegnato ad AWS DMS nel database SQL Server. Per ulteriori informazioni, consulta Ruoli a livello di database nella documentazione di SQL Server.
- Familiarità con l'uso delle `mongoimport` utilità `mongodumpmongorestore`,`mongoexport`, e per spostare dati da e verso un cluster Amazon DocumentDB.
- Microsoft SQL Server Management Studio, installato e configurato.

Limitazioni

- Il limite di dimensione del cluster in Amazon DocumentDB è di 64 TB. Per ulteriori informazioni, consulta Limiti del cluster nella documentazione di Amazon DocumentDB.
- AWS DMS non supporta l'unione di più tabelle di origine in un'unica raccolta Amazon DocumentDB.

- Se AWS DMS elabora modifiche da una tabella di origine senza una chiave primaria, ignorerà le colonne LOB (Large Object) nella tabella di origine.

Architecture

Stack tecnologico di origine

- Amazon EC2

Stack tecnologico Target

- Amazon DocumentDB

Architettura di destinazione

Tools (Strumenti)

- AWS DMS — AWS Database Migration Service (AWS DMS) ti aiuta a migrare i database in modo semplice e sicuro.
- Amazon DocumentDB — Amazon DocumentDB (con compatibilità MongoDB) è un servizio di database veloce, affidabile e completamente gestito.
- Amazon EC2 — Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS.
- Microsoft SQL Server: SQL Server è un sistema di gestione di database relazionali.
- SQL Server Management Studio (SSMS): SSMS è uno strumento per la gestione di SQL Server, incluso l'accesso, la configurazione e l'amministrazione dei componenti di SQL Server.

Epiche

Creare e configurare un VPC

Operazione	Description	Competenze richieste
Crea un VPC.	Accedi alla Console di gestione AWS e apri la console Amazon VPC. Crea	Amministratore di sistema

Operazione	Description	Competenze richieste
	un cloud privato virtuale (VPC) con un intervallo di blocchi IPv4 CIDR.	
Crea gruppi di sicurezza e rete. ACLs	Sulla console Amazon VPC, crea gruppi di sicurezza e liste di controllo degli accessi alla rete (rete ACLs) per il tuo VPC, in base alle tue esigenze. Puoi anche utilizzar e le impostazioni predefini te per queste configurazioni. Per ulteriori informazioni su questa e altre storie, consulta la sezione «Risorse correlate ».	Amministratore di sistema

Creare e configurare il cluster Amazon DocumentDB

Operazione	Description	Competenze richieste
Crea un cluster Amazon DocumentDB.	Apri la console Amazon DocumentDB e scegli «Clusters». Scegli «Crea» e crea un cluster Amazon DocumentDB con un'istanz a. Importante: assicurati di configurare questo cluster con i gruppi di sicurezza del tuo VPC.	Amministratore di sistema
Installa la shell mongo.	La shell mongo è un'utilità da riga di comando che si utilizza per connettersi al cluster Amazon DocumentD	Amministratore di sistema

Operazione	Description	Competenze richieste
	B e interrogarlo. Per installarlo, esegui il comando «/etc/yum.repos.d/mongodb-org-3.6.repo» per creare il file del repository. Esegui il comando «sudo yum install mongodb-org-shell -y" per installare la shell mongo. Per crittografare i dati in transito, scarica la chiave pubblica per Amazon DocumentDB, quindi connettiti alla tua istanza Amazon DocumentDB. Per ulteriori informazioni su questi passaggi, consulta la sezione «Risorse correlate».	
Crea un database nel cluster Amazon DocumentDB.	Esegui il comando «use» con il nome del tuo database per creare un database nel tuo cluster Amazon DocumentDB.	Amministratore di sistema

Crea e configura l'istanza di replica AWS DMS

Operazione	Description	Competenze richieste
Crea l'istanza di replica AWS DMS.	Apri la console AWS DMS e scegli «Crea istanza di replica». Inserisci un nome e una descrizione per l'attività di replica. Scegli la classe di istanza, la versione del motore, lo storage, il VPC, Multi-AZ e rendili accessibili al pubblico. Scegli la scheda	Amministratore di sistema

Operazione	Description	Competenze richieste
	«Avanzate» per configurare le impostazioni di rete e crittografia. Specificate le impostazioni di manutenzione, quindi scegliete «Crea istanza di replica».	
Configura il database SQL Server.	Accedi a Microsoft SQL Server e aggiungi una regola in entrata per la comunicazione tra l'endpoint di origine e l'istanza di replica AWS DMS. Utilizza l'indirizzo IP privato dell'istanza di replica come origine. Importante: l'istanza di replica e l'endpoint di destinazione devono trovarsi sullo stesso VPC. Utilizza una fonte alternativa nel gruppo di sicurezza se VPCs sono diverse per l'origine e le istanze di replica.	Amministratore di sistema

Crea e testa gli endpoint di origine e di destinazione in AWS DMS

Operazione	Description	Competenze richieste
Crea gli endpoint del database di origine e di destinazione.	Apri la console AWS DMS e scegli «Connetti gli endpoint del database di origine e di destinazione». Specificate le informazioni di connessione per i database di origine e di destinazione. Se necessario, scegli la scheda «Avanzate	Amministratore di sistema

Operazione	Description	Competenze richieste
	» per impostare i valori per «Attributi di connessione aggiuntivi». Scarica e usa il pacchetto di certificati nella configurazione del tuo endpoint.	
Verifica la connessione dell'endpoint.	Scegli «Esegui test» per testare la connessione. Risolvi eventuali messaggi di errore verificando le impostazioni del gruppo di sicurezza e le connessioni all'istanza di replica AWS DMS sia dall'istanza del database di origine che da quella di destinazione.	Amministratore di sistema

Migrazione dei dati

Operazione	Description	Competenze richieste
Crea l'attività di migrazione AWS DMS.	Nella console AWS DMS, scegli «Attività», «Crea attività». Specificate le opzioni dell'attività, inclusi i nomi degli endpoint di origine e di destinazione e i nomi delle istanze di replica. In «Tipo di migrazione» scegli «Migra i dati esistenti» e «Replica solo le modifiche ai dati». Scegli «Avvia attività».	Amministratore di sistema
Esegui l'attività di migrazione di AWS DMS.	In «Impostazioni attività», specifica le impostazioni per la	Amministratore di sistema

Operazione	Description	Competenze richieste
	modalità di preparazione della tabella, ad esempio «Non fare nulla», «Elimina tabelle sulla destinazione», «Truncate » e «Includi colonne LOB nella replica». Imposta una dimensione LOB massima che AWS DMS accetterà e scegli «Abilita registrazione». Lascia le «Impostazioni avanzate» ai valori predefiniti e scegli «Crea attività».	
Monitora la migrazione.	Nella console AWS DMS, scegli «Attività» e scegli l'attività di migrazione. Scegli «Task monitoring» per monitorare la tua attività. L'attività si interrompe quando la migrazione a pieno carico è completa e vengono applicate le modifiche memorizzate nella cache.	Amministratore di sistema

Testa e verifica la migrazione

Operazione	Description	Competenze richieste
Connettiti al cluster Amazon DocumentDB utilizzando la shell mongo.	Apri la console Amazon DocumentDB, scegli il tuo cluster in «Clusters». Nella scheda «Connettività e sicurezza», scegli «Connetti	Amministratore di sistema

Operazione	Description	Competenze richieste
	ti a questo cluster con la shell mongo».	
Verifica i risultati della migrazione.	Esegui il comando «use» con il nome del tuo database, quindi esegui il comando «show collections». Esegui il comando «db. .count ();» con il nome del tuo database. Se i risultati corrispondono al database di origine, la migrazione è avvenuta con successo.	Amministratore di sistema

Risorse correlate

Creare e configurare un VPC

- Crea un gruppo di sicurezza per il tuo VPC
- Crea un ACL di rete

Creare e configurare il cluster Amazon DocumentDB

- Crea un cluster Amazon DocumentDB
- Installa la shell mongo per Amazon DocumentDB
- Connettiti al tuo cluster Amazon DocumentDB

Crea e configura l'istanza di replica AWS DMS

- Usa istanze di replica pubbliche e private

Crea e testa gli endpoint di origine e di destinazione in AWS DMS

- Usa Amazon DocumentDB come destinazione per AWS DMS
- Usa un database SQL Server come origine per AWS DMS
- Usa gli endpoint AWS DMS

Migra i dati

- Migrazione ad Amazon DocumentDB

Altre risorse

- Limitazioni all'uso di SQL Server come sorgente per AWS DMS
- Come usare Amazon DocumentDB per creare e gestire applicazioni su larga scala

Esegui la migrazione di un database ThoughtSpot Falcon locale su Amazon Redshift

Battulga Purevragcha e Antony Prasad Thevaraj, Amazon Web Services

Riepilogo

I data warehouse locali richiedono tempi e risorse di amministrazione significativi, in particolare per set di dati di grandi dimensioni. Anche il costo finanziario della costruzione, della manutenzione e della crescita di questi magazzini è molto elevato. Per aiutare a gestire i costi, mantenere bassa la complessità di estrazione, trasformazione e caricamento (ETL) e fornire prestazioni man mano che i dati crescono, è necessario scegliere costantemente quali dati caricare e quali archiviare.

Migrando i database ThoughtSpot Falcon locali sul cloud Amazon Web Services (AWS), puoi accedere a data lake e data warehouse basati sul cloud che aumentano l'agilità aziendale, la sicurezza e l'affidabilità delle applicazioni, oltre a ridurre i costi complessivi dell'infrastruttura. Amazon Redshift aiuta a ridurre in modo significativo i costi e le spese operative di un data warehouse. Puoi anche utilizzare Amazon Redshift Spectrum per analizzare grandi quantità di dati nel suo formato nativo senza caricarli.

Questo modello descrive i passaggi e il processo per la migrazione di un database ThoughtSpot Falcon da un data center locale a un database Amazon Redshift sul cloud AWS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database ThoughtSpot Falcon ospitato in un data center locale

Versioni del prodotto

- ThoughtSpot versione 7.0.1

Architecture

Il diagramma mostra il flusso di lavoro seguente:

1. I dati sono ospitati in un database relazionale locale.

2. AWS Schema Conversion Tool (AWS SCT) converte il linguaggio di definizione dei dati (DDL) compatibile con Amazon Redshift.
3. Dopo aver creato le tabelle, puoi migrare i dati utilizzando AWS Database Migration Service (AWS DMS).
4. I dati vengono caricati in Amazon Redshift.
5. I dati vengono archiviati in Amazon Simple Storage Service (Amazon S3) se utilizzi Redshift Spectrum o se hai già ospitato i dati in Amazon S3.

Tools (Strumenti)

- AWS DMS: AWS Data Migration Service (AWS DMS) ti aiuta a migrare i database in modo rapido e sicuro su AWS.
- Amazon Redshift — Amazon Redshift è un servizio di data warehouse veloce, completamente gestito e su scala petabyte che semplifica ed economica l'analisi efficiente di tutti i dati utilizzando gli strumenti di business intelligence esistenti.
- AWS SCT — AWS Schema Conversion Tool (AWS SCT) converte lo schema di database esistente da un motore di database a un altro.

Epiche

Preparati per la migrazione

Operazione	Description	Competenze richieste
Identifica la configurazione Amazon Redshift appropriata.	Identifica la configurazione del cluster Amazon Redshift appropriata in base ai tuoi requisiti e al volume di dati. Per ulteriori informazioni, consulta i cluster Amazon Redshift nella documentazione di Amazon Redshift.	DBA

Operazione	Description	Competenze richieste
Cerca Amazon Redshift per valutare se soddisfa i tuoi requisiti.	Usa Amazon Redshift FAQs per capire e valutare se Amazon Redshift soddisfa i tuoi requisiti.	DBA

Prepara il cluster Amazon Redshift di destinazione

Operazione	Description	Competenze richieste
Crea un cluster Amazon Redshift.	Accedi alla Console di gestione AWS, apri la console Amazon Redshift e crea un cluster Amazon Redshift in un cloud privato virtuale (VPC). Per ulteriori informazioni, consulta Creazione di un cluster in un VPC nella documentazione di Amazon Redshift.	DBA
Esegui un PoC per la progettazione del tuo database Amazon Redshift.	Segui le best practice di Amazon Redshift eseguendo un proof of concept (PoC) per la progettazione del tuo database. Per ulteriori informazioni, consulta Condurre un proof of concept per Amazon Redshift nella documentazione di Amazon Redshift.	DBA
Crea utenti del database.	Crea gli utenti nel tuo database Amazon Redshift e	DBA

Operazione	Description	Competenze richieste
	concedi i ruoli appropriati per l'accesso allo schema e alle tabelle. Per ulteriori informazioni, consulta Concedere i privilegi di accesso per un utente o un gruppo di utenti nella documentazione di Amazon Redshift.	
Applica le impostazioni di configurazione al database di destinazione.	Applica le impostazioni di configurazione al database Amazon Redshift in base ai tuoi requisiti. Per ulteriori informazioni sull'abilitazione dei parametri a livello di database, sessione e server, consulta il riferimento alla configurazione nella documentazione di Amazon Redshift.	DBA

Crea oggetti nel cluster Amazon Redshift

Operazione	Description	Competenze richieste
Crea manualmente tabelle con DDL in Amazon Redshift.	(Facoltativo) Se utilizzi AWS SCT, le tabelle vengono create automaticamente. Tuttavia, se si verificano errori durante la replica DDLs, è necessario creare manualmente le tabelle	DBA

Operazione	Description	Competenze richieste
Crea tabelle esterne per Redshift Spectrum.	Crea una tabella esterna con uno schema esterno per Amazon Redshift Spectrum. Per creare tabelle esterne, devi essere il proprietario dello schema esterno o un superutente del database. Per ulteriori informazioni, consulta Creazione di tabelle esterne per Amazon Redshift Spectrum nella documenta zione di Amazon Redshift.	DBA

Migrazione dei dati con AWS DMS

Operazione	Description	Competenze richieste
Usa AWS DMS per migrare i dati.	Dopo aver creato il DDL delle tabelle nel database Amazon Redshift, migra i dati su Amazon Redshift utilizzan do AWS DMS. Per passaggi e istruzioni dettagliate, consulta Usare un database Amazon Redshift come destinazione per AWS DMS nella documentazione di AWS DMS.	DBA
Utilizzate il comando COPY per caricare i dati.	Usa il `COPY` comando Amazon Redshift per caricare i dati da Amazon S3 ad Amazon Redshift.	DBA

Operazione	Description	Competenze richieste
	Per ulteriori informazioni, consulta Utilizzo del comando COPY per il caricamento da Amazon S3 nella documentazione di Amazon Redshift.	

Convalida il cluster Amazon Redshift

Operazione	Description	Competenze richieste
Convalida i record di origine e di destinazione.	Convalida il conteggio delle tabelle per i record di origine e di destinazione che sono stati caricati dal sistema di origine.	DBA
Implementa le best practice di Amazon Redshift per l'ottimizzazione delle prestazioni.	Implementa le best practice di Amazon Redshift per la progettazione di tabelle e database. Per ulteriori informazioni, consulta il post sul blog Le 10 migliori tecniche di ottimizzazione delle prestazioni per Amazon Redshift.	DBA
Ottimizza le prestazioni delle query.	Amazon Redshift utilizza query basate su SQL per interagire con dati e oggetti nel sistema. Il linguaggio di manipolazione dei dati (DML) è il sottoinsieme di SQL che puoi utilizzare per visualizzare, aggiungere, modificare ed eliminare dati. DDL è il	DBA

Operazione	Description	Competenze richieste
	sottoinsieme di SQL utilizzato per aggiungere, modificare ed eliminare oggetti di database come tabelle e viste. Per ulteriori informazioni, consulta Tuning query performance nella documentazione di Amazon Redshift.	
Implementa WLM.	È possibile utilizzare la gestione del carico di lavoro (WLM) per definire più code di interrogazioni e indirizzare le query alle code appropriate in fase di esecuzione. Per ulteriori informazioni, consulta Implementazione della gestione del carico di lavoro nella documentazione di Amazon Redshift.	DBA
Lavora con il ridimensionamento simultaneo.	Utilizzando la funzionalità Concurrency Scaling, è possibile supportare un numero virtualmente illimitato di utenti e query simultanee, con prestazioni di query costantemente elevate. Per ulteriori informazioni, consulta Working with concurrency scaling nella documentazione di Amazon Redshift.	DBA

Operazione	Description	Competenze richieste
Utilizza le best practice di Amazon Redshift per la progettazione di tabelle.	Quando pianifichi il tuo database, alcune importanti decisioni sulla progettazione delle tabelle possono influenzare fortemente le prestazioni complessive delle query. Per ulteriori informazioni sulla scelta dell'opzione di progettazione delle tabelle più appropriata, consulta le best practice di Amazon Redshift per la progettazione di tabelle nella documentazione di Amazon Redshift.	DBA
Crea viste materializzate in Amazon Redshift.	Una vista materializzata contiene un set di risultati precalcolato basato su una query SQL su una o più tabelle di base. È possibile emettere `SELECT` istruzioni per interrogare una vista materializzata nello stesso modo in cui si esegue una query su altre tabelle o viste del database. Per ulteriori informazioni, consulta Creazione di viste materializzate in Amazon Redshift nella documentazione di Amazon Redshift.	DBA

Operazione	Description	Competenze richieste
Definire le giunzioni tra le tabelle.	Per cercare più di una tabella contemporaneamente ThoughtSpot, è necessari o definire i join tra le tabelle specificando le colonne che contengono i dati corrispon denti su due tabelle. Queste colonne rappresentano la fine `primary key` `foreign key` del join. Puoi definirli utilizzando il `ALTER TABLE` comando in Amazon Redshift o. ThoughtSpot Per ulteriori informazioni, consulta ALTER TABLE nella documentazione di Amazon Redshift.	DBA

Configura ThoughtSpot la connessione ad Amazon Redshift

Operazione	Description	Competenze richieste
Aggiungi una connessione Amazon Redshift.	Aggiungi una connessio ne Amazon Redshift al tuo database Falcon locale ThoughtSpot . Per ulteriori informazioni, consulta Aggiungere una connessione Amazon Redshift nella ThoughtSpot documenta zione.	DBA

Operazione	Description	Competenze richieste
Modifica la connessione Amazon Redshift.	Puoi modificare la connessio ne Amazon Redshift per aggiungere tabelle e colonne. Per ulteriori informazioni, consulta Modificare una connessione Amazon Redshift nella ThoughtSpot documenta zione.	DBA
Rimappa la connessione Amazon Redshift.	Modifica i parametri di connessione modifican do il file di mappatura dei sorgenti .yaml creato quando hai aggiunto la connessione Amazon Redshift. Ad esempio, puoi rimappare la tabella o la colonna esistente su una tabella o colonna diversa in una connessio ne al database esistente . ThoughtSpot consiglia di controllare le dipendenze prima e dopo aver rimappato una tabella o una colonna in una connessione per assicurar si che vengano visualizzate come richiesto. Per ulteriori informazioni, consulta Rimappare una connessione Amazon Redshift nella ThoughtSpot documenta zione.	DBA

Operazione	Description	Competenze richieste
Elimina una tabella dalla connessione Amazon Redshift.	(Facoltativo) Se tenti di rimuovere una tabella in una connessione Amazon Redshift, ThoughtSpot verifica le dipendenze e mostra un elenco di oggetti dipendenti. Puoi scegliere gli oggetti elencati per eliminarli o rimuovere la dipendenza. È quindi possibile rimuovere la tabella. Per ulteriori informazioni, consulta Eliminare una tabella da una connessione Amazon Redshift nella ThoughtSpot documentazione.	DBA
Elimina una tabella con oggetti dipendenti da una connessione Amazon Redshift.	(Facoltativo) Se tenti di eliminare una tabella con oggetti dipendenti, l'operazione viene bloccata. Viene visualizzata una `Cannot delete` finestra con un elenco di collegamenti agli oggetti dipendenti. Una volta rimosse tutte le dipendenze, è possibile eliminare la tabella Per ulteriori informazioni, consulta Eliminare una tabella con oggetti dipendenti da una connessione Amazon Redshift nella ThoughtSpot documentazione.	DBA

Operazione	Description	Competenze richieste
Elimina una connessione Amazon Redshift.	(Facoltativo) Poiché una connessione può essere utilizzata in più sorgenti di dati o visualizzazioni, è necessari o eliminare tutte le fonti e le attività che utilizzano tale connessione prima di poter eliminare la connessione Amazon Redshift. Per ulteriori informazioni, consulta Eliminare una connessione Amazon Redshift nella ThoughtSpot documenta zione.	DBA
Controlla il riferimento di connessione per Amazon Redshift.	Assicurati di fornire le informazioni richieste per la tua connessione Amazon Redshift utilizzando il riferimento Connection nella ThoughtSpot documentazione.	DBA

Informazioni aggiuntive

- Analisi basata sull'intelligenza artificiale su qualsiasi scala con Amazon ThoughtSpot Redshift
- Prezzi di Amazon Redshift
- Guida introduttiva ad AWS SCT
- Guida introduttiva ad Amazon Redshift
- Utilizzo di agenti di estrazione dei dati
- Chick-fil-A migliora la velocità di acquisizione delle informazioni con ThoughtSpot e AWS

Esegui la migrazione da Oracle Database ad Amazon RDS for PostgreSQL utilizzando Oracle GoldenGate

Dhairya Jindani, Sindhusha Paturu e Rajeshkumar Sabankar, Amazon Web Services

Riepilogo

Questo modello mostra come migrare un database Oracle su Amazon Relational Database Service (Amazon RDS) per PostgreSQL utilizzando Oracle Cloud Infrastructure (OCI). GoldenGate

Utilizzando Oracle GoldenGate, puoi replicare i dati tra il tuo database di origine e uno o più database di destinazione con tempi di inattività minimi.

> **Note**
>
> Il database Oracle di origine può essere locale o su un'istanza Amazon Elastic Compute Cloud EC2 (Amazon). È possibile utilizzare una procedura simile quando si utilizzano strumenti di replica locali.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Una GoldenGate licenza Oracle
- Driver Java Database Connectivity (JDBC) per la connessione al database PostgreSQL
- Schema e tabelle creati con AWS Schema Conversion Tool (AWS SCT) sul database Amazon RDS for PostgreSQL di destinazione

Limitazioni

- Oracle GoldenGate può replicare solo i dati delle tabelle esistenti (caricamento iniziale) e le modifiche in corso (acquisizione dei dati di modifica)

Versioni del prodotto

- Oracle Database Enterprise Edition 10g o versioni successive

- Oracle GoldenGate 12.2.0.1.1 per Oracle o versioni più recenti
- Oracle GoldenGate 12.2.0.1.1 per PostgreSQL o versioni più recenti

Architecture

Il diagramma seguente mostra un esempio di flusso di lavoro per la migrazione di un database Oracle ad Amazon RDS for PostgreSQL utilizzando Oracle: GoldenGate

Il diagramma mostra il flusso di lavoro seguente:

1. Il processo Oracle GoldenGate Extract viene eseguito sul database di origine per estrarre i dati.
2. Il processo Oracle GoldenGate Replicat fornisce i dati estratti al database Amazon RDS for PostgreSQL di destinazione.

Tools (Strumenti)

- Oracle ti GoldenGate aiuta a progettare, eseguire, orchestrare e monitorare la replica dei dati e le soluzioni di elaborazione dei dati in streaming nell'infrastruttura Oracle Cloud.
- Amazon Relational Database Service (Amazon RDS) per PostgreSQL ti aiuta a configurare, gestire e scalare un database relazionale PostgreSQL nel cloud AWS.

Epiche

Scarica e installa Oracle GoldenGate

Operazione	Description	Competenze richieste
Scarica Oracle GoldenGate.	Scarica le seguenti versioni di Oracle GoldenGate: • Oracle GoldenGate 12.2.0.1.1 per Oracle o una versione più recente • Oracle GoldenGate 12.2.0.1.1 per PostgreSQL o una versione più recente	DBA

Operazione	Description	Competenze richieste
	Per scaricare il software, consulta Oracle Downloads sul sito Web di Oracle. GoldenGate	
Installa Oracle GoldenGate for Oracle sul server Oracle Database di origine.	Per istruzioni, consulta la GoldenGate documentazione di Oracle.	DBA
Installa il database Oracle GoldenGate per PostgreSQL sull'istanza Amazon. EC2	Per istruzioni, consulta la documentazione di Oracle. GoldenGate	DBA

Configura Oracle GoldenGate sui database di origine e di destinazione

Operazione	Description	Competenze richieste
Configura Oracle GoldenGat e for Oracle Database sul database di origine.	Per istruzioni, consulta la GoldenGate documentazione di Oracle. Assicurati di configurare quanto segue: • Registrazione supplemen tare • Utenti Oracle GoldenGate • Eventuali concessioni e autorizzazioni necessarie • File di parametri • Processo di gestione • Directory • File GLOBALS • Wallet Oracle	DBA

Operazione	Description	Competenze richieste
Configura Oracle GoldenGate per PostgreSQL sul database di destinazione.	Per istruzioni, vedere la Parte VI Uso di Oracle GoldenGate per PostgreSQL sul sito Web di Oracle. Assicurati di configurare quanto segue: • Processo di gestione • File GLOBALS • Wallet Oracle	DBA

Configura l'acquisizione dei dati

Operazione	Description	Competenze richieste
Configura il processo di estrazione nel database di origine.	Nel database Oracle di origine, crea un file di estrazione per estrarre i dati. Per istruzioni, consulta ADD EXTRACT nella documenta zione di Oracle. **Note** Il file di estrazione include la creazione del file dei parametri di estrazione e della directory del file trail.	DBA
Configura una pompa di dati per trasferire il file trail	Crea un file di parametri EXTRACT e una directory di	DBA

Operazione	Description	Competenze richieste
dal database di origine al database di destinazione.	file trail seguendo le istruzioni in PARFILE in Database Utilities sul sito Web di Oracle. Per ulteriori informazioni, consulta What is a Trail? in Fusion Middleware Understanding Oracle GoldenGate sul sito Web di Oracle.	

Operazione	Description	Competenze richieste
Configura la replica sull' EC2 istanza Amazon.	Crea un file dei parametri di replica e una directory dei file trail. Per ulteriori informazioni sulla creazione di file di parametri di replica, vedere la sezione 3.5 Convalida di un file di parametri nella documenta zione di Oracle Database. Per ulteriori informazioni sulla creazione di una directory di file trail, vedere Creazione di un trail nella documentazione di Oracle Cloud. **Important** Assicurati di aggiunger e una voce della tabella dei checkpoin t nel file GLOBALS nella destinazione. Per ulteriori informazioni, consulta Cos'è un replicato? in Fusion Middleware Understan ding Oracle GoldenGate sul sito Web di Oracle.	DBA

Configurare la replica dei dati

Operazione	Description	Competenze richieste
Nel database di origine, create un file di parametri per estrarre i dati per il caricamento iniziale.	Segui le istruzioni in Creazione di un file di parametri in GGSCI nella documentazione di Oracle Cloud. **Important** Assicurati che il Manager sia in esecuzione sulla destinazione.	DBA
Nel database di destinazione, create un file di parametri per replicare i dati per il caricamento iniziale.	Segui le istruzioni in Creazione di un file di parametri in GGSCI nella documentazione di Oracle Cloud. **Important** Assicurati di aggiungere e avviare il processo Replicat.	DBA

Passare al database Amazon RDS for PostgreSQL

Operazione	Description	Competenze richieste
Interrompi il processo Replicat e assicurati che i database di	Confronta il numero di righe tra i database di origine e di destinazione per assicurarti	DBA

Operazione	Description	Competenze richieste
origine e di destinazione siano sincronizzati.	che la replica dei dati abbia avuto successo.	
Configura il supporto del linguaggio di definizione dei dati (DDL).	Esegui lo script DDL per creare trigger, sequenze, sinonimi e chiavi referenziali su PostgreSQL. **Note** È possibile utilizzare qualsiasi applicazione client SQL standard per connettersi a un database nel cluster DB. Ad esempio, puoi usare pgAdmin per connetterti alla tua istanza DB.	DBA

Risorse correlate

- Amazon RDS per PostgreSQL (Guida per l'utente di Amazon RDS)
- EC2 Documentazione Amazon
- Metodi e database di elaborazione GoldenGate supportati da Oracle (documentazione Oracle)

Esegui la migrazione di una tabella partizionata Oracle su PostgreSQL utilizzando AWS DMS

Saurav Mishra e Eduardo Valentim, Amazon Web Services

Riepilogo

Questo modello descrive come velocizzare il caricamento di una tabella partizionata da Oracle a PostgreSQL utilizzando AWS Database Migration Service (AWS DMS), che non supporta il partizionamento nativo. Il database PostgreSQL di destinazione può essere installato su Amazon Elastic Compute Cloud (Amazon) oppure può essere un'istanza DB Edition compatibile con EC2 Amazon PostgreSQL o Amazon Aurora Edition Service (Amazon RDS) per PostgreSQL.

Il caricamento di una tabella partizionata include i seguenti passaggi:

1. Crea una tabella principale simile alla tabella delle partizioni di Oracle, ma non include alcuna partizione.
2. Crea tabelle secondarie che erediteranno dalla tabella principale creata nel passaggio 1.
3. Create una funzione di procedura e un trigger per gestire gli inserti nella tabella principale.

Tuttavia, poiché il trigger viene attivato per ogni inserto, il caricamento iniziale con AWS DMS può essere molto lento.

Per velocizzare i caricamenti iniziali da Oracle a PostgreSQL 9.0, questo modello crea un task AWS DMS separato per ogni partizione e carica le tabelle secondarie corrispondenti. Si crea quindi un trigger durante il cutover.

La versione 10 di PostgreSQL supporta il partizionamento nativo. Tuttavia, in alcuni casi potresti decidere di utilizzare il partizionamento ereditato. Per ulteriori informazioni, vedere la sezione Informazioni aggiuntive.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database Oracle di origine con una tabella partizionata
- Un database PostgreSQL su AWS

Versioni del prodotto

- PostgreSQL 9.0

Architecture

Stack tecnologico di origine

- Una tabella partizionata in Oracle

Stack tecnologico Target

- Una tabella partizionata in PostgreSQL (su Amazon, Amazon RDS for PostgreSQL o Aurora PostgreSQL EC2)

Architettura Target

Tools (Strumenti)

- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali.

Epiche

Configura AWS DMS

Operazione	Description	Competenze richieste
Crea le tabelle in PostgreSQL.	Crea le tabelle principali e secondarie corrispondenti in PostgreSQL con le condizion i di controllo richieste per le partizioni.	DBA
Crea il task AWS DMS per ogni partizione.	Includi la condizione di filtro della partizione nel task AWS DMS. Mappare le partizion	DBA

Operazione	Description	Competenze richieste
	i alle tabelle secondarie PostgreSQL corrispondenti.	
Esegui le attività di AWS DMS utilizzando l'acquisizione dei dati a pieno carico e modifica (CDC).	Assicurati che il `StopTaskCachedChangesApplied` parametro sia impostato su `true` e che il `StopTaskCachedChangesNotApplied` parametro sia impostato su. `false`	DBA

Tagliare

Operazione	Description	Competenze richieste
Interrompi le attività di replica.	Prima di interrompere le attività, verificate che l'origine e la destinazione siano sincronizzate.	DBA
Crea un trigger nella tabella principale.	Poiché la tabella principale riceverà tutti i comandi di inserimento e aggiornamento, create un trigger che indirizzerà questi comandi alle rispettive tabelle secondarie in base alla condizione di partizionamento.	DBA

Risorse correlate

- AWS DMS
- Partizionamento delle tabelle (documentazione PostgreSQL)

Informazioni aggiuntive

Sebbene la versione 10 di PostgreSQL supporti il partizionamento nativo, potresti decidere di utilizzare il partizionamento ereditato per i seguenti casi d'uso:

- Il partizionamento impone una regola secondo cui tutte le partizioni devono avere lo stesso set di colonne del set principale, ma l'ereditarietà delle tabelle supporta i figli con colonne aggiuntive.
- L'ereditarietà delle tabelle supporta ereditarietà multiple.
- Il partizionamento dichiarativo supporta solo il partizionamento di elenchi e intervalli. Con l'ereditarietà delle tabelle, puoi dividere i dati come desideri. Tuttavia, se l'esclusione dei vincoli non è in grado di eliminare efficacemente le partizioni, le prestazioni delle query ne risentiranno.
- Alcune operazioni richiedono un blocco più forte quando si utilizza il partizionamento dichiarativo rispetto a quando si utilizza l'ereditarietà delle tabelle. Ad esempio, l'aggiunta o la rimozione di una partizione da o verso una tabella partizionata richiede un `ACCESS EXCLUSIVE` blocco sulla tabella principale, mentre un blocco è sufficiente per l'ereditarietà regolare. `SHARE UPDATE EXCLUSIVE`

Quando utilizzi partizioni di lavoro separate, puoi anche ricaricare le partizioni in caso di problemi di convalida di AWS DMS. Per un migliore controllo delle prestazioni e della replica, esegui attività su istanze di replica separate.

Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for MySQL

Jitender Kumar, Srini Ramaswamy e Neha Sharma, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni per la migrazione di un'istanza Amazon Relational Database Service (Amazon RDS) per Oracle DB a un'istanza DB Amazon RDS for MySQL su Amazon Web Services (AWS). Il modello utilizza AWS Database Migration Service (AWS DMS) e AWS Schema Conversion Tool (AWS SCT).

Il modello fornisce le migliori pratiche per gestire la migrazione delle procedure archiviate. Inoltre, illustra le modifiche al codice per supportare il livello applicativo.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un database sorgente Amazon RDS for Oracle.
- Un database di destinazione Amazon RDS for MySQL. I database di origine e di destinazione devono trovarsi nello stesso cloud privato virtuale (VPC). Se ne utilizzi più VPCs o devi disporre delle autorizzazioni di accesso richieste.
- Gruppi di sicurezza che consentono la connettività tra i database di origine e di destinazione, AWS SCT, il server delle applicazioni e AWS DMS.
- Un account utente con il privilegio richiesto per eseguire AWS SCT sul database di origine.
- Registrazione supplementare abilitata per l'esecuzione di AWS DMS sul database di origine.

Limitazioni

- Il limite di dimensione del database Amazon RDS di origine e destinazione è di 64 TB. Per informazioni sulle dimensioni di Amazon RDS, consulta la documentazione AWS.
- Oracle non fa distinzione tra maiuscole e minuscole per gli oggetti del database, ma MySQL no. AWS SCT può gestire questo problema durante la creazione di un oggetto. Tuttavia, è necessario un po' di lavoro manuale per supportare la totale indistinzione tra maiuscole e minuscole.
- Questa migrazione non utilizza le estensioni MySQL per abilitare le funzioni native di Oracle. AWS SCT gestisce la maggior parte della conversione, ma è necessario del lavoro per modificare il codice manualmente.

- Nell'applicazione sono necessarie modifiche al driver Java Database Connectivity (JDBC).

Versioni del prodotto

- Amazon RDS for Oracle 12.2.0.1 e versioni successive. Per le versioni di RDS for Oracle attualmente supportate, consulta la documentazione AWS.
- Amazon RDS for MySQL 8.0.15 e versioni successive. Per le versioni RDS for MySQL attualmente supportate, consulta la documentazione AWS.
- AWS DMS versione 3.3.0 e successive. Consulta la documentazione AWS per ulteriori informazioni sugli endpoint di origine e di destinazione supportati da AWS DMS.
- AWS SCT versione 1.0.628 e successive. Consulta la matrice di supporto degli endpoint di origine e destinazione di AWS SCT nella documentazione AWS.

Architecture

Stack tecnologico di origine

- Amazon RDS per Oracle. Per ulteriori informazioni, consulta Usare un database Oracle come sorgente per AWS DMS.

Stack tecnologico Target

- Amazon RDS per MySQL. Per ulteriori informazioni, consulta Usare un database compatibile con MySQL come destinazione per AWS DMS.

Architettura di migrazione

Nel diagramma seguente, AWS SCT copia e converte gli oggetti dello schema dal database di origine Amazon RDS for Oracle e invia gli oggetti al database di destinazione Amazon RDS for MySQL. AWS DMS replica i dati dal database di origine e li invia all'istanza Amazon RDS for MySQL.

Tools (Strumenti)

- AWS Data Migration Service ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali.

- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS. Questo modello utilizza Amazon RDS for Oracle e Amazon RDS for MySQL.
- AWS Schema Conversion Tool (AWS SCT) supporta migrazioni di database eterogenei convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione.

Epiche

Preparati per la migrazione

Operazione	Description	Competenze richieste
Convalida le versioni e i motori del database di origine e di destinazione.		DBA
Identifica i requisiti hardware per l'istanza del server di destinazione.		DBA, SysAdmin
Identifica i requisiti di archiviazione (tipo e capacità di archiviazione).		DBA, SysAdmin
Scegli il tipo di istanza corretto (capacità, funzionalità di archiviazione, funzionalità di rete).		DBA, SysAdmin
Identifica i requisiti di sicurezza dell'accesso alla rete per i database di origine e di destinazione.		DBA, SysAdmin
Scegli una strategia di migrazione delle applicazioni.	Valuta se preferisci un periodo di inattività totale o parziale per le attività successive.	DBA, proprietario dell'app SysAdmin

Configurare l'infrastruttura

Operazione	Description	Competenze richieste
Crea un VPC e delle sottoreti.		SysAdmin
Crea gruppi di sicurezza e liste di controllo degli accessi alla rete (). ACLs		SysAdmin
Configura e avvia l'istanza Amazon RDS for Oracle.		DBA, SysAdmin
Configura e avvia l'istanza Amazon RDS for MySQL.		DBA, SysAdmin
Preparare un test case per la convalida della conversione del codice.	Ciò contribuirà al test unitario del codice convertito.	DBA, Sviluppatore
Configura l'istanza AWS DMS.		
Configura gli endpoint di origine e destinazione in AWS DMS.		

Migrazione dei dati

Operazione	Description	Competenze richieste
Genera lo script del database di destinazione utilizzando AWS SCT.	Verifica l'accuratezza del codice convertito da AWS SCT. Sarà necessario un po' di lavoro manuale.	DBA, Sviluppatore
In AWS SCT, scegli l'imposta zione «Case Insensitive».	In AWS SCT, scegli Project Settings, Target Case Sensitivi ty, Case Insensitive.	DBA, Sviluppatore

Operazione	Description	Competenze richieste
In AWS SCT, scegli di non utilizzare la funzione nativa Oracle.	In Project Settings, controlla le funzioni TO_ CHAR/TO_NUMBER/TO _DATE.	DBA, Sviluppatore
Apporta modifiche al codice «sql%notfound».	Potrebbe essere necessario convertire il codice manualmente.	
Interrogazione su tabelle e oggetti nelle stored procedure (utilizzare query minuscole).		DBA, Sviluppatore
Crea lo script principale dopo aver apportato tutte le modifiche, quindi distribuisci lo script principale nel database di destinazione.		DBA, Sviluppatore
Esegui il test unitario delle procedure memorizzate e delle chiamate alle applicazioni utilizzando dati di esempio.		
Pulisci i dati creati durante il test unitario.		DBA, Sviluppatore
Elimina i vincoli di chiave esterna sul database di destinazione.	Questo passaggio è necessario per caricare i dati iniziali. Se non vuoi eliminare i vincoli di chiave esterna, devi creare un'attività di migrazione per i dati specifici delle tabelle primarie e secondarie.	DBA, Sviluppatore

Operazione	Description	Competenze richieste
Rilascia le chiavi primarie e le chiavi univoche sul database di destinazione.	Questo passaggio consente di ottenere prestazioni migliori per il carico iniziale.	DBA, Sviluppatore
Abilita la registrazione supplementare sul database di origine.		DBA
Crea un'attività di migrazion e per il carico iniziale in AWS DMS, quindi eseguila.	Scegli l'opzione per migrare i dati esistenti.	DBA
Aggiungi le chiavi primarie e le chiavi esterne al database di destinazione.	I vincoli devono essere aggiunti dopo il caricamento iniziale.	DBA, Sviluppatore
Crea un'attività di migrazione per la replica continua.	La replica continua mantiene il database di destinazione sincronizzato con il database di origine.	DBA

Migrazione delle applicazioni

Operazione	Description	Competenze richieste
Sostituisci le funzioni native di Oracle con le funzioni native di MySQL.		Proprietario dell'app
Assicurati che vengano utilizzati solo nomi in minuscolo per gli oggetti del database nelle query SQL.		DBA, proprietario dell'app SysAdmin

Passa al database di destinazione

Operazione	Description	Competenze richieste
Arresta il server delle applicazioni.		Proprietario dell'app
Verifica che i database di origine e di destinazione siano sincronizzati.		DBA, proprietario dell'app
Arresta l'istanza DB di Amazon RDS for Oracle.		DBA
Interrompi l'attività di migrazione.	Questa operazione si interromperà automaticamente dopo aver completato il passaggio precedente.	DBA
Cambia la connessione JDBC da Oracle a MySQL.		Proprietario dell'app, DBA
Avvia l'applicazione.		DBA SysAdmin, proprietario dell'app

Chiudi il progetto

Operazione	Description	Competenze richieste
Rivedi e convalida i documenti del progetto.		DBA, SysAdmin
Raccogli le metriche relative ai tempi di migrazione, alla percentuale di attività manuali rispetto a quelle eseguite con strumenti, ai risparmi sui costi, ecc.		DBA, SysAdmin

Operazione	Description	Competenze richieste
Blocca ed elimina le istanze AWS DMS.		DBA
Rimuovi gli endpoint di origine e di destinazione.		DBA
Rimuovi le attività di migrazione.		DBA
Scatta uno snapshot dell'istanza DB di Amazon RDS for Oracle.		DBA
Elimina l'istanza DB di Amazon RDS for Oracle.		DBA
Chiudi ed elimina tutte le altre risorse AWS temporanee che hai utilizzato.		DBA, SysAdmin
Chiudi il progetto e fornisci eventuali feedback.		DBA

Risorse correlate

- AWS DMS
- AWS SCT
- Prezzi di Amazon RDS
- Guida introduttiva ad AWS DMS
- Nozioni di base su Amazon RDS

Esegui la migrazione da IBM Db2 su Amazon a EC2 Aurora PostgreSQL compatibile con AWS DMS e AWS SCT

Sirsendu Halder e Abhimanyu Chhabra, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni per la migrazione di un database IBM Db2 su un'istanza Amazon Elastic Compute Cloud (Amazon EC2) verso un'istanza DB Edition compatibile con Amazon Aurora PostgreSQL. Questo modello utilizza AWS Database Migration Service (AWS DMS) e AWS Schema Conversion Tool (AWS SCT) per la migrazione dei dati e la conversione dello schema.

Il modello mira a una strategia di migrazione online con tempi di inattività minimi o nulli per un database IBM Db2 da più terabyte con un numero elevato di transazioni. Ti consigliamo di convertire le colonne in chiavi primarie (PKs) e chiavi esterne (FKs) con il tipo di dati `BIGINT` in `INT` o in PostgreSQL `NUMERIC` per prestazioni migliori.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database IBM Db2 di origine su un'istanza EC2

Versioni del prodotto

- DB2/LINUXX8664 versione 11.1.4.4 e successive

Architecture

Stack tecnologico di origine

- Un database Db2 su un'istanza EC2

Stack tecnologico Target

- Un'istanza DB Aurora compatibile con PostgreSQL versione 10.18 o successiva

Architettura di migrazione del database

Tools (Strumenti)

- AWS Database Migration Service (AWS DMS) ti aiuta a migrare i database nel cloud AWS o tra combinazioni di configurazioni cloud e locali. Il database di origine rimane pienamente operativo durante la migrazione, riducendo al minimo i tempi di inattività delle applicazioni che si basano sul database. Puoi utilizzare AWS DMS per migrare i tuoi dati da e verso i database commerciali e open source più utilizzati. AWS DMS supporta migrazioni eterogenee tra diverse piattaforme di database, come IBM Db2 verso Aurora PostgreSQL versione 10.18 o superiore. Per i dettagli, consulta Sources for Data Migration e Targets for Data Migration nella documentazione di AWS DMS.
- AWS Schema Conversion Tool (AWS SCT) supporta migrazioni di database eterogenee convertendo automaticamente lo schema del database di origine e la maggior parte degli oggetti del codice del database, tra cui viste, stored procedure e funzioni, in un formato compatibile con il database di destinazione. Tutti gli oggetti che non vengono convertiti automaticamente sono chiaramente contrassegnati in modo che possano essere convertiti manualmente per completare la migrazione. AWS SCT può anche scansionare il codice sorgente dell'applicazione alla ricerca di istruzioni SQL incorporate e convertirle.

Epiche

Configura l'ambiente

Operazione	Description	Competenze richieste
Crea un'istanza DB compatibile con Aurora PostgreSQL.	Per creare l'istanza DB, segui le istruzioni nella documentazione AWS. Per il tipo di motore, scegli Amazon Aurora. Per l'edizione, scegli l'edizione compatibile con Amazon Aurora PostgreSQL. L'istanza DB Aurora compatibile con PostgreSQL versione 10.18 o successiva deve trovarsi nello stesso cloud	Amazon RDS

Operazione	Description	Competenze richieste
	privato virtuale (VPC) del database IBM Db2 di origine.	

Converti lo schema del tuo database

Operazione	Description	Competenze richieste
Installa e verifica AWS SCT.	1. Installa AWS SCT seguendo i passaggi nella documentazione di AWS SCT. 2. Verifica l'installazione seguendo le procedure nella documentazione di AWS SCT.	Amministratore AWS, DBA, ingegnere addetto alla migrazione
Avvia AWS SCT e crea un progetto.	Per avviare lo strumento AWS SCT e creare un nuovo progetto per eseguire un rapporto di valutazione della migrazione del database, segui le istruzioni nella documentazione di AWS SCT.	Ingegnere della migrazione
Aggiungi server di database e crea una regola di mappatura.	1. Aggiungi server di database di origine e di destinazione seguendo le istruzioni nella documentazione di AWS SCT. 2. Crea una regola di mappatura per definire la piattaforma di database di destinazione per il tuo database di origine. Per	Ingegnere della migrazione

Operazione	Description	Competenze richieste
	istruzioni, consulta la documentazione di AWS SCT.	
Crea un rapporto di valutazione della migrazione del database.	Crea il report di valutazione della migrazione del database seguendo i passaggi nella documentazione di AWS SCT.	Ingegnere della migrazione
Visualizza il rapporto di valutazione.	Utilizza la scheda Riepilogo del rapporto di valutazione della migrazione del database per visualizzare il rapporto e analizzare i dati. Questa analisi ti aiuterà a determinare la complessità della migrazione. Per ulteriori informazioni, consulta la documentazione di AWS SCT.	Ingegnere della migrazione

Operazione	Description	Competenze richieste
Convertire lo schema.	Per convertire gli schemi del database di origine: 1. Nella console AWS SCT, scegli Visualizza, quindi Visualizzazione principale. 2. Seleziona l'oggetto o il nodo principale dallo schema di origine, apri il menu contestuale (fai clic con il pulsante destro del mouse), quindi scegli Converti schema. Per ulteriori informazioni, consulta la documentazione di AWS SCT.	Ingegnere della migrazione
Applica lo schema del database convertito all'istanza DB di destinazione.	1. Scegli l'elemento dello schema nel pannello destro del tuo progetto che mostra lo schema pianificato per la tua istanza database di destinazione; 2. Apri il menu contestuale (tasto destro del mouse) per l'elemento dello schema e quindi scegli Apply to database (Applica al database). Per ulteriori informazioni, consulta la documentazione di AWS SCT.	Ingegnere della migrazione

Migra i tuoi dati

Operazione	Description	Competenze richieste
Configura un gruppo di parametri VPC e DB.	Configura un gruppo di parametri VPC e DB e configura le regole e i parametri in entrata necessari per la migrazione. Per istruzioni, consulta la documentazione di AWS DMS. Per il gruppo di sicurezza VPC, seleziona sia l' EC2 istanza per Db2 che l'istanza DB Aurora compatibile con PostgreSQL. Questa istanza di replica deve trovarsi nello stesso VPC delle istanze DB di origine e di destinazione.	Ingegnere della migrazione
Prepara le istanze DB di origine e di destinazione.	Prepara le istanze DB di origine e di destinazione per la migrazione. In un ambiente di produzione, il database di origine esisterà già. Per il database di origine, il nome del server deve essere il Domain Name System (DNS) pubblico dell' EC2 istanza in cui è in esecuzione Db2. Come nome utente, è possibile utilizzare `db2inst1` seguito dalla porta, che sarà 5000 per IBM Db2.	Ingegnere della migrazione

Operazione	Description	Competenze richieste
Crea un EC2 client Amazon e degli endpoint.	1. Crea un EC2 client Amazon. Utilizzi questo client per popolare il database di origine con dati da replicare. Il client viene utilizzato anche per verificar e la replica eseguendo le query sul database di destinazione. 2. Crea endpoint per il database di origine e l'istanza DB di destinazione da utilizzare per i passaggi successivi. Per istruzion i, consulta la documenta zione di AWS DMS. È necessario creare endpoint separati per i database di origine e di destinazione. Per la versione 10.18 o successiva compatibile con Aurora PostgreSQL, la porta sarà 5432 ed è possibile ottenere il nome del server dall'endpoint dell'istanza DB.	Ingegnere della migrazione

Operazione	Description	Competenze richieste
Crea un'istanza di replica.	Crea un'istanza di replica utilizzando la console AWS DMS e specifica gli endpoint di origine e destinazione. L'istanza di replica esegue la migrazione dei dati tra gli endpoint. Per ulteriori informazioni, consulta la documentazione di AWS DMS.	Ingegnere della migrazione

Operazione	Description	Competenze richieste
Crea un'attività AWS DMS per migrare i dati.	Crea un'attività per caricare le tabelle IBM Db2 di origine nell'istanza database PostgreSQL di destinazione seguendo i passaggi nella documentazione di AWS DMS. • Per l'origine e la destinazione, utilizza i nomi degli endpoint di origine e destinazione. • Il tipo di migrazione può essere a pieno carico. • Per la regola dello schema, è possibile utilizzare lo `inst1` schema del database Db2. • Per il nome della tabella, specificare di `%` migrare tutte le tabelle. Una volta completato il caricamento, vedrai le tabelle Db2 `inst1` dello schema apparire nel database Aurora compatibile con PostgreSQL.	Ingegnere della migrazione

Risorse correlate

Riferimenti

- Documentazione Amazon Aurora
- Documentazione FDW (Foreign Data Wrapper) di PostgreSQL
- Documentazione PostgreSQL IMPORT FOREIGN SCHEMA

- Documentazione AWS DMS
- Documentazione AWS SCT

Tutorial e video

- Guida introduttiva ad AWS DMS (procedura dettagliata)
- Introduzione ad Amazon EC2 - Elastic Cloud Server e hosting con AWS (video)

Esegui la migrazione da Oracle 8i o 9i ad Amazon RDS for PostgreSQL utilizzando AWS DMS SharePlex

Kumar Babu PG, Amazon Web Services

Riepilogo

Questo modello descrive come migrare un database Oracle 8i o 9i locale su Amazon Relational Database Service (Amazon RDS) per PostgreSQL o Amazon Aurora PostgreSQL. AWS Database Migration Service (AWS DMS) non supporta Oracle 8i o 9i come origine, quindi Quest SharePlex replica i dati da un database 8i o 9i locale a un database Oracle intermedio (Oracle 10g o 11g), compatibile con AWS DMS.

Dall'istanza Oracle intermedia, lo schema e i dati vengono migrati al database PostgreSQL su AWS utilizzando AWS Schema Conversion Tool (AWS SCT) e AWS DMS. Questo metodo consente di ottenere uno streaming continuo di dati dal database Oracle di origine all'istanza database PostgreSQL di destinazione con un ritardo di replica minimo. In questa implementazione, il downtime è limitato al tempo necessario per creare o convalidare tutte le chiavi esterne, i trigger e le sequenze sul database PostgreSQL di destinazione.

La migrazione utilizza un'istanza Amazon Elastic Compute Cloud (Amazon EC2) con Oracle 10g o 11g installato per ospitare le modifiche dal database Oracle di origine. AWS DMS utilizza questa istanza Oracle intermedia come origine per lo streaming dei dati verso Amazon RDS for PostgreSQL o Aurora PostgreSQL. La replica dei dati può essere messa in pausa e ripresa dal database Oracle locale all'istanza Oracle intermedia. Può anche essere messo in pausa e ripreso dall'istanza Oracle intermedia al database PostgreSQL di destinazione in modo da poter convalidare i dati utilizzando la convalida dei dati AWS DMS o uno strumento di convalida dei dati personalizzato.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database Oracle 8i o 9i di origine in un data center locale
- AWS Direct Connect configurato tra il data center locale e AWS
- Driver Java Database Connectivity (JDBC) per i connettori AWS SCT installati su un computer locale o sull'istanza EC2 in cui è installato AWS SCT
- Familiarità con l'uso di un database Oracle come sorgente AWS DMS

- Familiarità con l'uso di un database PostgreSQL come target AWS DMS
- Familiarità con la replica dei dati Quest SharePlex

Limitazioni

- Il limite di dimensione del database è di 64 TB
- Il database Oracle locale deve essere Enterprise Edition

Versioni del prodotto

- Oracle 8i o 9i per il database di origine
- Oracle 10g o 11g per il database intermedio
- PostgreSQL 9.6 o versione successiva

Architecture

Stack tecnologico di origine

- Database Oracle 8i o 9i
- Quest SharePlex

Stack tecnologico Obiettivo

- Amazon RDS per PostgreSQL o Aurora PostgreSQL

Architettura di origine e destinazione

Tools (Strumenti)

- AWS DMS — AWS Database Migration Service (AWS DMS) ti aiuta a migrare i database in modo rapido e sicuro. Il database di origine rimane pienamente operativo durante la migrazione, riducendo al minimo i tempi di inattività delle applicazioni che si basano sul database. AWS DMS può migrare i dati da e verso i database commerciali e open source più utilizzati.

- AWS SCT — AWS Schema Conversion Tool (AWS SCT) rende prevedibili le migrazioni di database eterogenei convertendo automaticamente lo schema del database di origine e la maggior parte degli oggetti del codice del database, tra cui viste, stored procedure e funzioni, in un formato compatibile con il database di destinazione. Gli oggetti che non possono essere convertiti automaticamente sono chiaramente contrassegnati in modo che possano essere convertiti manualmente per completare la migrazione. AWS SCT può anche scansionare il codice sorgente dell'applicazione alla ricerca di istruzioni SQL incorporate e convertirle come parte di un progetto di conversione dello schema del database. Durante questo processo, AWS SCT esegue l'ottimizzazione del codice nativo del cloud convertendo le funzioni legacy di Oracle e SQL Server nelle loro equivalenti AWS, per aiutarti a modernizzare le tue applicazioni durante la migrazione dei database. Una volta completata la conversione dello schema, AWS SCT può aiutare a migrare i dati da una serie di data warehouse ad Amazon Redshift utilizzando agenti di migrazione dei dati integrati.
- Quest SharePlex — Quest SharePlex è uno strumento di replica Oracle-to-Oracle dei dati per spostare i dati con tempi di inattività minimi e nessuna perdita di dati.

Epiche

Crea l' EC2 istanza e installa Oracle

Operazione	Description	Competenze richieste
Configura la rete per Amazon EC2.	Crea il cloud privato virtuale (VPC), le sottoreti, il gateway Internet, le tabelle di routing e i gruppi di sicurezza.	AWS SysAdmin
Crea la nuova EC2 istanza.	Seleziona l'Amazon Machine Image (AMI) per l' EC2 istanza. Scegli la dimension e dell'istanza e configura i dettagli dell'istanza: il numero di istanze (1), il VPC e la sottorete del passaggio precedente, assegnazione automatica dell'IP pubblico e altre opzioni. Aggiungi spazio	AWS SysAdmin

Operazione	Description	Competenze richieste
	di archiviazione, configura i gruppi di sicurezza e avvia l'istanza. Quando richiesto, create e salvate una key pair per il passaggio successivo.	
Installa Oracle sull' EC2 istanza.	Acquisisci le licenze e i file binari Oracle richiesti e installa Oracle 10g o 11g sull'istanza. EC2	DBA

Configura SharePlex su un' EC2 istanza e configura la replica dei dati

Operazione	Description	Competenze richieste
Configurare SharePlex.	Crea un' EC2 istanza Amazon e installa i SharePlex file binari compatibili con Oracle 8i o 9i.	AWS SysAdmin, DBA
Configura la replica dei dati.	Segui le SharePlex best practice per configurare la replica dei dati da un database Oracle 8i/9i locale a un'istanza Oracle 10g/11g.	DBA

Convertire lo schema del database Oracle in PostgreSQL

Operazione	Description	Competenze richieste
Configura AWS SCT.	Crea un nuovo report, quindi connettiti a Oracle come origine e PostgreSQL come destinazione. Nelle impostazi oni del progetto, apri la scheda	DBA

Operazione	Description	Competenze richieste
	SQL Scripting e modifica lo script SQL di destinazione in File multipli.	
Convertire lo schema del database Oracle.	Nella scheda Azione, scegli Genera report, Converti schema e quindi Salva come SQL.	DBA
Modifica gli script SQL generati da AWS SCT.		DBA

Crea e configura l'istanza database Amazon RDS

Operazione	Description	Competenze richieste
Crea l'istanza database Amazon RDS.	Nella console Amazon RDS, crea una nuova istanza DB PostgreSQL.	AWS SysAdmin, DBA
Configura l'istanza DB.	Specificate la versione del motore DB, la classe dell'istanza DB, l'implementazione Multi-AZ, il tipo di storage e lo storage allocato. Immettere l'identificatore dell'istanza DB, un nome utente principale e una password principale.	AWS SysAdmin, DBA
Configura rete e sicurezza.	Specificare il VPC, il gruppo di sottoreti, l'accessibilità pubblica, la preferenza della zona di disponibilità e i gruppi di sicurezza.	AWS SysAdmin, DBA

Operazione	Description	Competenze richieste
Configura le opzioni del database.	Specificare il nome del database, la porta, il gruppo di parametri, la crittografia e la chiave principale.	AWS SysAdmin, DBA
Configurare i backup.	Specificate il periodo di conservazione del backup, la finestra di backup, l'ora di inizio, la durata e se copiare i tag nelle istantanee.	AWS SysAdmin, DBA
Configura le opzioni di monitoraggio.	Abilita o disabilita il monitoraggio avanzato e le informazioni sulle prestazioni.	AWS SysAdmin, DBA
Configura le opzioni di manutenzione.	Specificare l'aggiornamento automatico della versione secondaria, la finestra di manutenzione e il giorno, l'ora e la durata di inizio.	AWS SysAdmin, DBA
Esegui gli script di pre-migrazione da AWS SCT.	Nell'istanza Amazon RDS, esegui questi script: create_database.sql, create_sequence.sql, create_table.sql, create_view.sql e create_function.sql.	AWS SysAdmin, DBA

Migra i dati utilizzando AWS DMS

Operazione	Description	Competenze richieste
Crea un'istanza di replica in AWS DMS.	Completa i campi per il nome, la classe dell'istanza, il VPC (come per l' EC2 istanza),	AWS SysAdmin, DBA

Operazione	Description	Competenze richieste
	Multi-AZ e l'accessibilità pubblica. Nella sezione di configurazione avanzata, specifica lo storage allocato, il gruppo di sottoreti, la zona di disponibilità, i gruppi di sicurezza VPC e la chiave principale di AWS Key Management Service (AWS KMS).	
Crea l'endpoint del database di origine.	Specificare il nome dell'endpoint, il tipo, il motore di origine (Oracle), il nome del server (nome DNS EC2 privato Amazon), la porta, la modalità SSL, il nome utente, la password, il SID, il VPC (specifica il VPC che ha l'istanza di replica) e l'istanza di replica. Per testare la connessione, scegli Run Test, quindi crea l'endpoint. Puoi anche configurare le seguenti impostazioni avanzate: maxFileSize e numberDataType Scale.	AWS SysAdmin, DBA

Operazione	Description	Competenze richieste
Crea l'attività di replica AWS DMS.	Specificare il nome dell'attività, l'istanza di replica, gli endpoint di origine e di destinazione e l'istanza di replica. Per il tipo di migrazione, scegli «Migra i dati esistenti e replica le modifiche in corso». Deselezio na la casella di controllo «Avvia attività al momento della creazione».	AWS SysAdmin, DBA
Configura le impostazioni delle attività di replica di AWS DMS.	Per la modalità di preparazi one della tabella di destinazi one, scegli «Non fare nulla». Interrompi l'attività dopo il completamento del caricamen to completo per creare le chiavi primarie. Specifica te la modalità LOB limitata o completa e attivate le tabelle di controllo. Facoltati vamente, è possibile configura re l'impostazione CommitRate avanzata.	DBA

Operazione	Description	Competenze richieste
Configura le mappature delle tabelle.	Nella sezione mappature delle tabelle, crea una regola di inclusione per tutte le tabelle in tutti gli schemi inclusi nella migrazione, quindi crea una regola di esclusione. Aggiungi tre regole di trasformazione per convertire i nomi di schema, tabella e colonna in lettere minuscole e aggiungi tutte le altre regole necessari e per questa migrazione specifica.	DBA
Avvia l’attività .	Avvia l'attività di replica. Accertatevi che il carico sia in esecuzione a pieno carico. Esegui ALTER SYSTEM SWITCH LOGFILE sul database Oracle primario per avviare l'attività.	DBA
Esegui gli script di migrazione intermedia da AWS SCT.	In Amazon RDS for PostgreSQL, esegui questi script: create_index.sql e create_constraint.sql.	DBA
Riavviare l'attività per continuare l'acquisizione dei dati delle modifiche (CDC).	Nell'istanza DB Amazon RDS for PostgreSQL, esegui VACUUM e riavvia l'attività AWS DMS per applicare le modifiche CDC memorizzate nella cache.	DBA

Passa al database PostgreSQL

Operazione	Description	Competenze richieste
Controlla i log e le tabelle dei metadati di AWS DMS.	Convalida eventuali errori e correggili se necessario.	DBA
Interrompi tutte le dipendenze Oracle.	Chiudi i listener sul database Oracle ed esegui ALTER SYSTEM SWITCH LOGFILE. Interrompi l'attività AWS DMS quando non mostra alcuna attività.	DBA
Esegui gli script post-migr azione da AWS SCT.	In Amazon RDS for PostgreSQL, esegui questi script: create_foreign_key _constraint.sql e create_tr iggers.sql.	DBA
Completa eventuali passaggi aggiuntivi di Amazon RDS for PostgreSQL.	Incrementa le sequenze in modo che corrispon dano a quelle di Oracle, se necessario, esegui VACUUM e ANALYZE e scatta un'istantanea per verificare la conformità.	DBA
Apri le connessioni ad Amazon RDS for PostgreSQL.	Rimuovi i gruppi di sicurezza AWS DMS da Amazon RDS for PostgreSQL, aggiungi gruppi di sicurezza di produzione e indirizza le tue applicazioni verso il nuovo database.	DBA

Operazione	Description	Competenze richieste
Pulisci le risorse AWS DMS.	Rimuovi gli endpoint, le attività di replica, le istanze di replica e l'istanza. EC2	SysAdmin, DBA

Risorse correlate

- Documentazione AWS DMS
- Documentazione AWS SCT
- Prezzi di Amazon RDS per PostgreSQL
- Utilizzo di un database Oracle come origine per AWS DMS
- Utilizzo di un database PostgreSQL come destinazione per AWS DMS
- Documentazione Quest SharePlex

Esegui la migrazione da Oracle 8i o 9i ad Amazon RDS for PostgreSQL utilizzando viste materializzate e AWS DMS

Kumar Babu PG e Pragnesh Patel, Amazon Web Services

Riepilogo

Questo modello descrive come migrare un database Oracle 8i o 9i legacy locale verso Amazon Relational Database Service (Amazon RDS) per PostgreSQL o Amazon Aurora PostgreSQL Compatible Edition.

AWS Database Migration Service (AWS DMS) non supporta Oracle 8i o 9i come sorgente, quindi questo modello utilizza un'istanza di database Oracle intermedia compatibile con AWS DMS, come Oracle 10g o 11g. Utilizza anche la funzionalità di viste materializzate per migrare i dati dall'istanza Oracle 8i/9i di origine all'istanza Oracle 10g/11g intermedia.

AWS Schema Conversion Tool (AWS SCT) converte lo schema del database e AWS DMS migra i dati nel database PostgreSQL di destinazione.

Questo modello aiuta gli utenti che desiderano migrare dai database Oracle legacy con tempi di inattività minimi del database. In questa implementazione, il tempo di inattività sarebbe limitato al tempo necessario per creare o convalidare tutte le chiavi, i trigger e le sequenze esterne sul database di destinazione.

Il modello utilizza istanze Amazon Elastic Compute Cloud (Amazon EC2) con un database Oracle 10g/11g installato per aiutare AWS DMS a trasmettere i dati. Puoi sospendere temporaneamente la replica in streaming dal database Oracle locale all'istanza Oracle intermedia per consentire ad AWS DMS di recuperare il ritardo sulla convalida dei dati o di utilizzare un altro strumento di convalida dei dati. L'istanza DB PostgreSQL e il database Oracle intermedio avranno gli stessi dati quando AWS DMS avrà terminato la migrazione delle modifiche correnti.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database Oracle 8i o 9i di origine in un data center locale
- AWS Direct Connect configurato tra il data center locale e AWS
- Driver Java Database Connectivity (JDBC) per i connettori AWS SCT installati su un computer locale o sull'istanza EC2 in cui è installato AWS SCT

- Familiarità con l'uso di un database Oracle come sorgente AWS DMS
- Familiarità con l'uso di un database PostgreSQL come target AWS DMS

Limitazioni

- Il limite di dimensione del database è di 64 TB

Versioni del prodotto

- Oracle 8i o 9i per il database di origine
- Oracle 10g o 11g per il database intermedio
- PostgreSQL 10.17 o versione successiva

Architecture

Stack tecnologico di origine

- Database Oracle 8i o 9i

Stack tecnologico Target

- Compatibile con Amazon RDS per PostgreSQL o Aurora PostgreSQL

Architettura Target

Tools (Strumenti)

- AWS DMS aiuta a migrare i database in modo rapido e sicuro. Il database di origine rimane pienamente operativo durante la migrazione, riducendo al minimo i tempi di inattività delle applicazioni che si basano sul database. AWS DMS può migrare i tuoi dati da e verso i database commerciali e open source più utilizzati.
- AWS SCT converte automaticamente lo schema del database di origine e la maggior parte degli oggetti del codice del database, incluse viste, stored procedure e funzioni, in un formato compatibile con il database di destinazione. Gli oggetti che non possono essere convertiti automaticamente sono contrassegnati in modo chiaro in modo che possano essere convertiti manualmente per completare la migrazione. AWS SCT può anche scansionare il codice sorgente

dell'applicazione alla ricerca di istruzioni SQL incorporate e convertirle come parte di un progetto di conversione dello schema del database. Durante questo processo, AWS SCT esegue l'ottimizzazione del codice nativo del cloud convertendo le funzioni legacy di Oracle e SQL Server nelle loro equivalenti AWS, per aiutarti a modernizzare le tue applicazioni durante la migrazione dei database. Una volta completata la conversione dello schema, AWS SCT può aiutare a migrare i dati da una serie di data warehouse ad Amazon Redshift utilizzando agenti di migrazione dei dati integrati.

Best practice

Per le best practice per l'aggiornamento delle viste materializzate, consulta la seguente documentazione Oracle:

- Aggiornamento delle viste materializzate
- Aggiornamento rapido per le viste materializzate

Epiche

Installa Oracle su un' EC2 istanza e crea viste materializzate

Operazione	Description	Competenze richieste
Configura la rete per l' EC2 istanza.	Crea il cloud privato virtuale (VPC), le sottoreti, il gateway Internet, le tabelle di routing e i gruppi di sicurezza.	AWS SysAdmin
Crea l' EC2 istanza.	Seleziona l'Amazon Machine Image (AMI) per l' EC2 istanza. Scegli la dimension e dell'istanza e configura i dettagli dell'istanza: il numero di istanze (1), il VPC e la sottorete del passaggio precedente, assegnazione automatica dell'IP pubblico e altre opzioni. Aggiungi spazio	AWS SysAdmin

Operazione	Description	Competenze richieste
	di archiviazione, configura i gruppi di sicurezza e avvia l'istanza. Quando richiesto, create e salvate una key pair per il passaggio successivo.	
Installa Oracle sull' EC2 istanza.	Acquisisci le licenze e i file binari Oracle richiesti e installa Oracle 10g o 11g sull'istanza. EC2	DBA
Configurare la rete Oracle.	Modifica o aggiungi voci `listener.ora` per connetterti al database Oracle 8i/9i di origine locale, quindi crea i collegamenti al database.	DBA
Crea viste materializzate.	Identifica gli oggetti del database da replicare nel database Oracle 8i/9i di origine, quindi crea viste materializzate per tutti gli oggetti utilizzando il database link.	DBA
Implementa script per aggiornare le viste materiali zzate agli intervalli richiesti.	Sviluppa e distribuisci script per aggiornare le viste materializzate a intervalli richiesti sull'istanza Amazon EC2 Oracle 10g/11g. Utilizza l'opzione di aggiornamento incrementale per aggiornare le viste materializzate.	DBA

Convertire lo schema del database Oracle in PostgreSQL

Operazione	Description	Competenze richieste
Configura AWS SCT.	Crea un nuovo report, quindi connettiti a Oracle come origine e PostgreSQL come destinazione. Nelle impostazi oni del progetto, apri la scheda SQL Scripting. Cambia lo script SQL di destinazione in File multipli. (AWS SCT non supporta i database Oracle 8i/9i, quindi è necessario ripristinare il dump basato solo sullo schema sull'istanza Oracle 10g/11g intermedia e utilizzarlo come sorgente per AWS SCT.)	DBA
Convertire lo schema del database Oracle.	Nella scheda Azione, scegli Genera report, Converti schema e quindi Salva come SQL.	DBA
Modifica gli script SQL.	Apporta modifiche in base alle migliori pratiche. Ad esempio, passa a tipi di dati adatti e sviluppa equivalen ti PostgreSQL per funzioni specifiche di Oracle.	DBA, devDBA

Crea e configura l'istanza DB di Amazon RDS per ospitare il database convertito

Operazione	Description	Competenze richieste
Crea l'istanza database Amazon RDS.	Nella console Amazon RDS, crea una nuova istanza DB PostgreSQL.	AWS SysAdmin, DBA
Configura l'istanza DB.	Specificate la versione del motore DB, la classe dell'istanza DB, l'implementazione Multi-AZ, il tipo di storage e lo storage allocato. Immettere l'identificatore dell'istanza DB, un nome utente principale e una password principale.	AWS SysAdmin, DBA
Configura rete e sicurezza.	Specificare il VPC, il gruppo di sottoreti, l'accessibilità pubblica, la preferenza della zona di disponibilità e i gruppi di sicurezza.	DBA, SysAdmin
Configurare le opzioni del database.	Specificare il nome del database, la porta, il gruppo di parametri, la crittografia e la chiave principale.	DBA, AWS SysAdmin
Configurare i backup.	Specificate il periodo di conservazione del backup, la finestra di backup, l'ora di inizio, la durata e se copiare i tag nelle istantanee.	AWS SysAdmin, DBA
Configura le opzioni di monitoraggio.	Abilita o disabilita il monitoraggio avanzato e gli approfondimenti sulle prestazioni.	AWS SysAdmin, DBA

Operazione	Description	Competenze richieste
Configura le opzioni di manutenzione.	Specificare l'aggiornamento automatico della versione secondaria, la finestra di manutenzione e il giorno, l'ora e la durata di inizio.	AWS SysAdmin, DBA
Esegui gli script di pre-migrazione da AWS SCT.	Sull'istanza Amazon RDS for PostgreSQL di destinazione, crea lo schema del database utilizzando gli script SQL di AWS SCT con altre modifiche. Queste potrebbero includere l'esecuzione di più script e includere la creazione di utenti, la creazione di database, la creazione di schemi, tabelle, viste, funzioni e altri oggetti di codice.	AWS SysAdmin, DBA

Migra i dati utilizzando AWS DMS

Operazione	Description	Competenze richieste
Crea un'istanza di replica in AWS DMS.	Completa i campi per il nome, la classe dell'istanza, il VPC (come per l' EC2 istanza), Multi-AZ e l'accessibilità pubblica. Nella sezione di configurazione avanzata, specifica lo storage allocato, il gruppo di sottoreti, la zona di disponibilità, i gruppi di sicurezza VPC e la chiave	AWS SysAdmin, DBA

Operazione	Description	Competenze richieste
	AWS Key Management Service (AWS KMS).	
Crea l'endpoint del database di origine.	Specificare il nome dell'endpoint, il tipo, il motore di origine (Oracle), il nome del server (il nome DNS privato dell'EC2 istanza), la porta, la modalità SSL, il nome utente, la password, il SID, il VPC (specificare il VPC che ha l'istanza di replica) e l'istanza di replica. Per testare la connessione, scegli Esegui test, quindi crea l'endpoint. Puoi anche configurare le seguenti impostazioni avanzate: maxFileSizee numberDataTypeScale.	AWS SysAdmin, DBA
Connetti AWS DMS ad Amazon RDS per PostgreSQL.	Crea un gruppo di sicurezza per la migrazione per le connessioni VPCs, se il tuo database PostgreSQL si trova in un altro VPC.	AWS SysAdmin, DBA

Operazione	Description	Competenze richieste
Crea l'endpoint del database di destinazione.	Specificare il nome dell'endpoint, il tipo, il motore di origine (PostgreSQL), il nome del server (endpoint Amazon RDS), la porta, la modalità SSL, il nome utente, la password, il nome del database, il VPC (specifica il VPC che contiene l'istanza di replica) e l'istanza di replica. Per testare la connessione, scegli Esegui test, quindi crea l'endpoint. Puoi anche configurare le seguenti impostazioni avanzate: maxFileSizee numberDataTypeScale.	AWS SysAdmin, DBA
Crea l'attività di replica AWS DMS.	Specificare il nome dell'attività, l'istanza di replica, gli endpoint di origine e di destinazione e l'istanza di replica. Per il tipo di migrazione, scegli Migra i dati esistenti e replica le modifiche in corso. Deselezionа la casella di controllo Avvia attività alla creazione.	AWS SysAdmin, DBA

Operazione	Description	Competenze richieste
Configura le impostazioni delle attività di replica di AWS DMS.	Per la modalità di preparazione della tabella di destinazione, scegli Non fare nulla. Interrompi l'operazione al termine del caricamento completo (per creare le chiavi primarie). Specifica te la modalità LOB limitata o completa e attivate le tabelle di controllo. Facoltativamente, è possibile configurare l'impostazione CommitRateavanzata.	DBA
Configura le mappature delle tabelle.	Nella sezione Mappature delle tabelle, crea una regola di inclusione per tutte le tabelle in tutti gli schemi inclusi nella migrazione, quindi crea una regola di esclusione. Aggiungi tre regole di trasformazione per convertire i nomi di schema, tabella e colonna in lettere minuscole e aggiungi tutte le altre regole necessarie per questa migrazione specifica.	DBA
Avvia l'attività .	Avvia l'attività di replica. Accertatevi che il carico sia in esecuzione a pieno carico. Esegui `ALTER SYSTEM SWITCH LOGFILE` sul database Oracle primario per avviare il task.	DBA

Operazione	Description	Competenze richieste
Esegui gli script di migrazione intermedia da AWS SCT.	In Amazon RDS for PostgreSQL, esegui i `create_index.sql` seguenti `create_constraint.sql` script: e (se lo schema completo non è stato inizialmente creato).	DBA
Riprendi l'attività per continuare l'acquisizione dei dati delle modifiche (CDC).	Esegui `VACUUM` sull'istanza DB Amazon RDS for PostgreSQL e riavvia l'attività AWS DMS per applicare le modifiche CDC memorizzate nella cache.	DBA

Passa al database PostgreSQL

Operazione	Description	Competenze richieste
Controlla i log e le tabelle di convalida di AWS DMS.	Controlla e correggi eventuali errori di replica o convalida.	DBA
Smetti di usare il database Oracle locale e le sue dipendenze.	Arresta tutte le dipendenze Oracle, spegni i listener sul database Oracle ed esegui. `ALTER SYSTEM SWITCH LOGFILE` Interrompi l'attività AWS DMS quando non mostra alcuna attività.	DBA
Esegui gli script post-migrazione da AWS SCT.	In Amazon RDS for PostgreSQL, esegui questi script:. `create_foreign_key_constraint.sql and create_tr`	DBA

Operazione	Description	Competenze richieste
	`iggers.sql` Assicurati che le sequenze siano aggiornate.	
Completa i passaggi aggiuntivi di Amazon RDS for PostgreSQL.	Incrementa le sequenze in modo che corrispondano a quelle di Oracle, se necessari o, esegui `VACUUM` e scatta un'istantanea per `ANALYZE` garantire la conformità.	DBA
Apri le connessioni ad Amazon RDS for PostgreSQL.	Rimuovi i gruppi di sicurezza AWS DMS da Amazon RDS for PostgreSQL, aggiungi gruppi di sicurezza di produzione e indirizza le tue applicazioni verso il nuovo database.	DBA
Pulisci gli oggetti AWS DMS.	Rimuovi gli endpoint, le attività di replica, le istanze di replica e l'istanza. EC2	SysAdmin, DBA

Risorse correlate

- Documentazione AWS DMS
- Documentazione AWS SCT
- Prezzi di Amazon RDS per PostgreSQL
- Utilizzo di un database Oracle come origine per AWS DMS
- Utilizzo di un database PostgreSQL come destinazione per AWS DMS

Esegui la migrazione da Oracle su Amazon EC2 ad Amazon RDS for MySQL utilizzando AWS DMS e AWS SCT

Anil Kunapareddy, Amazon Web Services

Harshad Gohil, Nessuno

Riepilogo

La gestione dei database Oracle sulle istanze Amazon Elastic Compute Cloud (Amazon EC2) richiede risorse e può essere costosa. Lo spostamento di questi database su un'istanza database Amazon Relational Database Service (Amazon RDS) per MySQL semplificherà il tuo lavoro ottimizzando il budget IT complessivo. Amazon RDS for MySQL offre anche funzionalità come Multi-AZ, scalabilità e backup automatici.

Questo modello illustra la migrazione di un database Oracle di origine su Amazon EC2 verso un'istanza database Amazon RDS for MySQL di destinazione. Utilizza AWS Database Migration Service (AWS DMS) per migrare i dati e AWS Schema Conversion Tool (AWS SCT) per convertire lo schema e gli oggetti del database di origine in un formato compatibile con Amazon RDS for MySQL.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database di origine con servizi di istanza e listener in esecuzione, in modalità ARCHIVELOG
- Un database Amazon RDS for MySQL di destinazione, con spazio di archiviazione sufficiente per la migrazione dei dati

Limitazioni

- AWS DMS non crea uno schema sul database di destinazione; devi farlo. Il nome dello schema deve già esistere per la destinazione. Le tabelle dello schema di origine vengono importate in user/schema, che AWS DMS utilizza per connettersi all'istanza di destinazione. Per migrare più schemi, devi creare più attività di replica.

Versioni del prodotto

- Tutte le edizioni del database Oracle per le versioni 10.2 e successive, 11g e fino a 12.2 e 18c. Per l'elenco più recente delle versioni supportate, consulta Utilizzo di un database Oracle come origine per AWS DMS e Utilizzo di un database compatibile con MySQL come destinazione per AWS DMS. Ti consigliamo di utilizzare la versione più recente di AWS DMS per il supporto più completo della versione e delle funzionalità. Per informazioni sulle versioni dei database Oracle supportate da AWS SCT, consulta la documentazione di AWS SCT.
- AWS DMS supporta le versioni 5.5, 5.6 e 5.7 di MySQL.

Architecture

Stack tecnologico di origine

- Un database Oracle su un'istanza EC2

Stack tecnologico Target

- Istanza database Amazon RDS per MySQL

Architettura di migrazione dei dati

Architettura di origine e destinazione

Tools (Strumenti)

- AWS DMS - AWS Database Migration Service (AWS DMS) è un servizio Web che puoi utilizzare per migrare i dati dal tuo database locale, su un'istanza DB Amazon RDS o in un database su un'istanza EC2, verso un database su un servizio AWS come Amazon RDS for MySQL o un'istanza EC2. Puoi anche migrare un database da un servizio AWS a un database locale. È possibile migrare i dati tra motori di database eterogenei o omogenei.
- AWS SCT - AWS Schema Conversion Tool (AWS SCT) rende prevedibili le migrazioni di database eterogenei convertendo automaticamente lo schema del database di origine e la maggior parte degli oggetti del codice del database, tra cui viste, stored procedure e funzioni, in un formato compatibile con il database di destinazione. Dopo aver convertito lo schema del database e gli oggetti di codice utilizzando AWS SCT, puoi utilizzare AWS DMS per migrare i dati dal database di origine al database di destinazione per completare i tuoi progetti di migrazione.

Epiche

Pianifica la migrazione

Operazione	Description	Competenze richieste
Identifica le versioni e i motori del database di origine e di destinazione.		DBA/Sviluppatore
Identifica l'istanza di replica DMS.		DBA/Sviluppatore
Identifica i requisiti di archiviazione come il tipo e la capacità di archiviazione.		DBA/Sviluppatore
Identifica i requisiti di rete come latenza e larghezza di banda.		DBA/Sviluppatore
Identifica i requisiti hardware per le istanze del server di origine e di destinazione (in base all'elenco di compatibilità e ai requisiti di capacità di Oracle).		DBA/Sviluppatore
Identifica i requisiti di sicurezza dell'accesso alla rete per i database di origine e di destinazione.		DBA/Sviluppatore
Installa i driver AWS SCT e Oracle.		DBA/Sviluppatore
Determina una strategia di backup.		DBA/Sviluppatore

Operazione	Description	Competenze richieste
Determina i requisiti di disponibilità.		DBA/Sviluppatore
Identifica la strategia di migrazione e commutazione delle applicazioni.		DBA/Sviluppatore
Seleziona il tipo di istanza DB corretto in base alla capacità, allo storage e alle funzionalità di rete.		DBA/Sviluppatore

Configura l'ambiente

Operazione	Description	Competenze richieste
Crea un cloud privato virtuale (Virtual Private Cloud (VPC)) L'origine, la destinazione e l'istanza di replica devono trovarsi nello stesso VPC. È inoltre utile averle nella stessa zona di disponibilità.		Developer
Crea i gruppi di sicurezza necessari per l'accesso al database.		Developer
Genera e configura una key pair.		Developer
Configura sottoreti, zone di disponibilità e blocchi CIDR.		Developer

Configura l'origine: database Oracle sull'istanza EC2

Operazione	Description	Competenze richieste
Installa Oracle Database su Amazon EC2 con gli utenti e i ruoli richiesti.		DBA
Esegui i tre passaggi nella colonna successiva per accedere a Oracle dall'esterno dell'istanza EC2.	1. Cambia l'host locale nel `tnsnames` DNS pubblico di Amazon EC2. 2. Cambia l'host locale nel `listener` DNS pubblico di Amazon EC2. 3. Arresta e riavvia il listener.	DBA
Quando Amazon EC2 viene riavviato, il DNS pubblico cambia. Assicurati di aggiornare il DNS pubblico di Amazon EC2 in 'tnsnames' e 'listener' o usa un indirizzo IP elastico.		DBA/Sviluppatore
Configura il gruppo di sicurezza dell'istanza EC2 in modo che l'istanza di replica e i client richiesti possano accedere al database di origine.		DBA/Sviluppatore

Configurare la destinazione: Amazon RDS for MySQL

Operazione	Description	Competenze richieste
Configura e avvia l'istanza DB Amazon RDS for MySQL.		Developer

Operazione	Description	Competenze richieste
Crea il tablespace necessario nell'istanza database Amazon RDS for MySQL.		DBA
Configurare il gruppo di sicurezza in modo che l'istanza di replica e i client richiesti possano accedere al database di destinazione.		Developer

Configura AWS SCT e crea uno schema nel database di destinazione

Operazione	Description	Competenze richieste
Installa i driver AWS SCT e Oracle.		Developer
Inserisci i parametri appropriati e connettiti all'origine e alla destinazione.		Developer
Genera un rapporto di conversione dello schema.		Developer
Se necessario, correggete il codice e lo schema, in particolare tablespace e virgolette, ed eseguiteli sul database di destinazione.		Developer
Convalida lo schema sull'origine rispetto alla destinazione prima di migrare i dati.		Developer

Migrazione dei dati con AWS DMS

Operazione	Description	Competenze richieste
Per l'acquisizione di dati a pieno carico e modifica (CDC) o solo per CDC, è necessari o impostare un attributo di connessione aggiuntivo.		Developer
All'utente specificato nelle definizioni del database Oracle di origine AWS DMS devono essere concessi tutti i privilegi richiesti. Per un elenco completo, consulta https://docs.aws.amazon.com /dms/ latest/userguide/C HAP _source.oracle.html #CHAP_Source .Oracle.Self-Managed.		DBA/Sviluppatore
Abilita la registrazione supplementare nel database di origine.		DBA/Sviluppatore
Per l'acquisizione di dati a pieno carico e modifica (CDC) o solo per CDC, abilita la modalità ARCHIVELOG nel database di origine.		DBA
Crea endpoint di origine e destinazione e testa le connessioni.		Developer

Operazione	Description	Competenze richieste
Quando gli endpoint sono collegati correttamente, crea un'attività di replica.		Developer
Seleziona solo CDC (o) a pieno carico più CDC nell'atti vità per acquisire le modifiche per la sola replica continua (o) a pieno carico più le modifiche in corso, rispettivamente.		Developer
Esegui l'attività di replica e monitora i CloudWatch log di Amazon.		Developer
Convalida i dati nei database di origine e di destinazione.		Developer

Migra la tua applicazione e taglia

Operazione	Description	Competenze richieste
Segui i passaggi per la tua strategia di migrazione delle applicazioni.		DBA, sviluppatore, proprietario dell'app
Segui i passaggi per la tua strategia di cutover/switch-ove r delle applicazioni.		DBA, sviluppatore, proprietario dell'app

Chiudi il progetto

Operazione	Description	Competenze richieste
Convalida lo schema e i dati nei database di origine e di destinazione.		DBA/Sviluppatore
Raccogli le metriche in base alle tempistiche necessari e per la migrazione, la percentuale di utilizzo manuale rispetto agli strumenti , i risparmi sui costi, ecc.		DBA/Developer/AppOwner
Esamina i documenti e gli artefatti del progetto.		DBA/Developer/AppOwner
Chiudi le risorse AWS temporanee.		DBA/Sviluppatore
Chiudi il progetto e fornisci feedback.		DBA/Developer/AppOwner

Risorse correlate

- Documentazione AWS DMS
- Sito web AWS DMS
- Post sul blog di AWS DMS
- Strategie di migrazione di Oracle Database su AWS
- Amazon RDS per Oracle FAQs
- Domande frequenti Oracle
- Amazon EC2
- Amazon EC2 FAQs
- Licenza del software Oracle nell'ambiente di cloud computing

Esegui la migrazione di un database Oracle da Amazon EC2 ad Amazon RDS for MariaDB utilizzando AWS DMS e AWS SCT

Veeranjaneyulu Grandhi e vinod kumar, Amazon Web Services

Riepilogo

Questo modello illustra i passaggi per la migrazione di un database Oracle su un'istanza Amazon Elastic Compute Cloud (Amazon EC2) a un'istanza Amazon Relational Database Service (Amazon RDS) per MariaDB. Il modello utilizza AWS Data Migration Service (AWS DMS) per la migrazione dei dati e AWS Schema Conversion Tool (AWS SCT) per la conversione dello schema.

La gestione dei database Oracle sulle EC2 istanze richiede più risorse ed è più costosa rispetto all'utilizzo di un database su Amazon RDS. Amazon RDS semplifica la configurazione, il funzionamento e la scalabilità di un database relazionale nel cloud. Amazon RDS offre una capacità ridimensionabile e conveniente, automatizzando al contempo attività amministrative dispendiose in termini di tempo come il provisioning dell'hardware, la configurazione del database, l'applicazione di patch e i backup.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un database Oracle di origine con servizi di istanza e listener attivi e funzionanti. Questo database deve essere in modalità ARCHIVELOG.
- Familiarità con l'utilizzo di un database Oracle come sorgente per AWS DMS.
- Familiarità con l'uso di Oracle come fonte per AWS SCT.

Limitazioni

- Limite di dimensione del database: 64 TB

Versioni del prodotto

- Tutte le edizioni del database Oracle per le versioni 10.2 e successive, 11g e fino a 12.2 e 18c. Per l'elenco più recente delle versioni supportate, consulta Using an Oracle Database as a Source for AWS DMS e la tabella delle versioni di AWS SCT nella documentazione AWS.

- Amazon RDS supporta le versioni 10.3, 10.4, 10.5 e 10.6 di MariadB Server Community Server. Per l'elenco più recente delle versioni supportate, consulta la documentazione di Amazon RDS.

Architecture

Stack tecnologico di origine

- Un database Oracle su un'istanza EC2

Stack tecnologico Target

- Amazon RDS per MariaDB

Architettura di migrazione dei dati

Architettura Target

Tools (Strumenti)

- AWS Schema Conversion Tool (AWS SCT) rende prevedibili le migrazioni di database eterogenei convertendo automaticamente lo schema del database di origine e la maggior parte degli oggetti del codice del database, tra cui viste, stored procedure e funzioni, in un formato compatibile con il database di destinazione. Dopo aver convertito lo schema del database e gli oggetti di codice utilizzando AWS SCT, puoi utilizzare AWS DMS per migrare i dati dal database di origine al database di destinazione per completare i tuoi progetti di migrazione. Per ulteriori informazioni, consulta Using Oracle as a Source for AWS SCT nella documentazione di AWS SCT.
- AWS Database Migration Service (AWS DMS) ti aiuta a migrare i database in AWS in modo rapido e sicuro. Il database di origine rimane pienamente operativo durante la migrazione, riducendo al minimo i tempi di inattività delle applicazioni che si basano sul database. AWS DMS può migrare i tuoi dati da e verso i database commerciali e open source più utilizzati. AWS DMS supporta migrazioni omogenee da Oracle a Oracle, nonché migrazioni eterogenee tra diverse piattaforme di database, come Oracle o Microsoft SQL Server verso Amazon Aurora. Per ulteriori informazioni sulla migrazione dei database Oracle, consulta Using an Oracle Database as a Source for AWS DMS nella documentazione di AWS DMS.

Epiche

Piano per la migrazione

Operazione	Description	Competenze richieste
Identifica le versioni e i motori di database.	Identifica le versioni e i motori del database di origine e di destinazione.	DBA, Sviluppatore
Identifica l'istanza di replica.	Identifica l'istanza di replica AWS DMS.	DBA, sviluppatore
Identifica i requisiti di archiviazione.	Identifica il tipo e la capacità di storage.	DBA, sviluppatore
Identifica i requisiti di rete.	Identifica la latenza e la larghezza di banda della rete.	DBA, sviluppatore
Identifica i requisiti hardware.	Identifica i requisiti hardware per le istanze del server di origine e di destinazione (in base all'elenco di compatibilità e ai requisiti di capacità di Oracle).	DBA, sviluppatore
Identifica i requisiti di sicurezza.	Identifica i requisiti di sicurezza dell'accesso alla rete per i database di origine e di destinazione.	DBA, Sviluppatore
Installa i driver.	Installa i driver AWS SCT e Oracle più recenti.	DBA, Sviluppatore
Determina una strategia di backup.		DBA, Sviluppatore
Determina i requisiti di disponibilità.		DBA, Sviluppatore

Operazione	Description	Competenze richieste
Scegli una migration/switchover strategia applicativa.		DBA, Sviluppatore
Selezionare il tipo di istanza .	Seleziona il tipo di istanza corretto in base alla capacità, allo storage e alle funzionalità di rete.	DBA, Sviluppatore

Configura l'ambiente

Operazione	Description	Competenze richieste
Crea un cloud privato virtuale (Virtual Private Cloud (VPC))	Le istanze di origine, destinazione e replica devono trovarsi nello stesso VPC e nella stessa zona di disponibilità (scelta consigliata).	Developer
Crea gruppi di sicurezza.	Creare i gruppi di sicurezza necessari per l'accesso al database.	Developer
Genera una coppia di chiavi.	Genera e configura una key pair.	Developer
Configura altre risorse.	Configura sottoreti, zone di disponibilità e blocchi CIDR.	Developer

Configura la fonte

Operazione	Description	Competenze richieste
Avvia l' EC2 istanza.	Per istruzioni, consulta la EC2 documentazione di Amazon.	Developer

Operazione	Description	Competenze richieste
Installa il database Oracle.	Installa il database Oracle sull' EC2 istanza, con gli utenti e i ruoli richiesti.	DBA
Segui i passaggi indicati nella descrizione del task per accedere a Oracle dall'esterno dell' EC2 istanza.	1. Cambia l'host locale nel `tnsnames` DNS EC2 pubblico di Amazon. 2. Cambia l'host locale nel `listener` DNS EC2 pubblico di Amazon. 3. Arresta e riavvia il listener.	DBA
Aggiorna il DNS EC2 pubblico di Amazon.	Dopo il riavvio dell' EC2 istanza, il DNS pubblico cambia. Assicurati di aggiornare il DNS EC2 pubblico di Amazon in `tnsnames elistener`, oppure usa un indirizzo IP elastico.	DBA, Sviluppatore
Configura il gruppo di sicurezza dell' EC2 istanza.	Configura il gruppo di sicurezza dell' EC2 istanza in modo che l'istanza di replica e i client richiesti possano accedere al database di origine.	DBA, Sviluppatore

Configura l'ambiente Amazon RDS for MariaDB di destinazione

Operazione	Description	Competenze richieste
Avvia l'istanza DB RDS.	Configura e avvia l'istanza DB Amazon RDS for MariaDB.	Developer

Operazione	Description	Competenze richieste
Crea tablespace.	Crea tutti i tablespace necessari nel database Amazon RDS MariaDB.	DBA
Configurare un gruppo di sicurezza.	Configura un gruppo di sicurezza in modo che l'istanza di replica e i client richiesti possano accedere al database di destinazione.	Developer

Configurazione di AWS SCT

Operazione	Description	Competenze richieste
Installa i driver.	Installa i driver AWS SCT e Oracle più recenti.	Developer
Connect (Connetti).	Inserisci i parametri appropriati e poi connettiti all'origine e alla destinazione.	Developer
Genera un rapporto di conversione dello schema.	Genera un report di conversio ne dello schema AWS SCT.	Developer
Correggi il codice e lo schema secondo necessità.	Apportate le correzion i necessarie al codice e allo schema (in particolare tablespace e virgolette).	DBA, Sviluppatore
Convalidare lo schema.	Convalida lo schema sull'orig ine rispetto alla destinazione prima di caricare i dati.	Developer

Migrazione dei dati con AWS DMS

Operazione	Description	Competenze richieste
Imposta un attributo di connessione.	Per l'acquisizione dei dati a pieno carico e modifica (CDC) o solo per CDC, imposta un attributo di connessio ne aggiuntivo. Per ulteriori informazioni, consulta la documentazione di Amazon RDS.	Developer
Abilita la registrazione supplementare.	Abilita la registrazione supplementare sul database di origine.	DBA, Sviluppatore
Abilita la modalità di registro di archiviazione.	Per il caricamento completo e il CDC (o solo per CDC), abilita la modalità di registro di archiviazione sul database di origine.	DBA
Crea e testa gli endpoint.	Crea endpoint di origine e destinazione e testa le connessioni. Per ulteriori informazioni, consulta la documentazione di Amazon DMS.	Developer
Crea un'attività di replica.	Quando gli endpoint sono collegati correttamente, crea un'attività di replica. Per ulteriori informazioni, consulta la documentazione di Amazon DMS.	Developer

Operazione	Description	Competenze richieste
Scegli il tipo di replica.	Scegli CDC only o Full load plus CDC nell'attività di acquisizione delle modifiche solo per la replica continua o rispettivamente per le modifiche a pieno carico e in corso.	Developer
Avvia e monitora l'attività.	Avvia l'attività di replica e monitora i CloudWatch log di Amazon. Per ulteriori informazioni, consulta la documentazione di Amazon DMS.	Developer
Convalida i dati.	Convalida i dati nei database di origine e di destinazione.	Developer

Migra le applicazioni e trasferiscile al database di destinazione

Operazione	Description	Competenze richieste
Segui la strategia di migrazione e delle applicazioni scelta.		DBA, proprietario dell'app, sviluppatore
Segui la cutover/switchover strategia applicativa scelta.		DBA, proprietario dell'app, sviluppatore

Chiudi il progetto

Operazione	Description	Competenze richieste
Convalida lo schema e i dati.	Assicurati che lo schema e i dati siano convalidati correttamente nell'origine	DBA, Sviluppatore

Operazione	Description	Competenze richieste
	rispetto alla destinazione prima della chiusura del progetto.	
Raccogli le metriche.	Raccogli le metriche relative al tempo di migrazione, alla percentuale di attività manuali rispetto a quelle eseguite con l'utensile, al risparmio sui costi e a criteri simili.	DBA, proprietario dell'app, sviluppatore
Consulta la documentazione.	Esamina i documenti e gli artefatti del progetto.	DBA, proprietario dell'app, sviluppatore
Chiudi le risorse.	Chiudi le risorse AWS temporanee.	DBA, Sviluppatore
Chiudi il progetto.	Chiudi il progetto di migrazion e e fornisci eventuali feedback.	DBA, proprietario dell'app, sviluppatore

Risorse correlate

- Panoramica di MariaDB Amazon RDS
- Dettagli del prodotto Amazon RDS for MariaDB
- Utilizzo di un database Oracle come sorgente per AWS DMS
- Strategie per la migrazione dei database Oracle su AWS
- Licenza del software Oracle nell'ambiente di cloud computing
- Amazon RDS per Oracle FAQs
- Panoramica di AWS DMS
- Post sul blog di AWS DMS
- EC2 Panoramica di Amazon
- Amazon EC2 FAQs
- Documentazione AWS SCT

Esegui la migrazione di un database Oracle locale ad Amazon RDS for MySQL utilizzando AWS DMS e AWS SCT

Sergey Dmitriev e Naresh Damera, Amazon Web Services

Riepilogo

Questo modello illustra la migrazione di un database Oracle locale a un'istanza DB Amazon Relational Database Service (Amazon RDS) per MySQL. Utilizza AWS Database Migration Service (AWS DMS) per migrare i dati e AWS Schema Conversion Tool (AWS SCT) per convertire lo schema e gli oggetti del database di origine in un formato compatibile con Amazon RDS for MySQL.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database Oracle di origine in un data center locale

Limitazioni

- Limite di dimensione del database: 64 TB

Versioni del prodotto

- Tutte le edizioni del database Oracle per le versioni 11g (versioni 11.2.0.3.v1 e successive) e fino a 12.2 e 18c. Per l'elenco più recente delle versioni supportate, consulta Using an Oracle Database as a Source for AWS DMS. Ti consigliamo di utilizzare la versione più recente di AWS DMS per il supporto più completo della versione e delle funzionalità. Per informazioni sulle versioni dei database Oracle supportate da AWS SCT, consulta la documentazione di AWS SCT.
- AWS DMS attualmente supporta le versioni di MySQL 5.5, 5.6 e 5.7. Per l'elenco più recente delle versioni supportate, consulta Using a MySQL compatible Database as a Target for AWS DMS nella documentazione AWS.

Architecture

Stack tecnologico di origine

- Database Oracle locale

Stack tecnologico Target

- Istanza database Amazon RDS per MySQL

Architettura di migrazione dei dati

Tools (Strumenti)

- AWS DMS - AWS Database Migration Services (AWS DMS) ti aiuta a migrare database relazionali, data warehouse, database NoSQL e altri tipi di archivi dati. Puoi utilizzare AWS DMS per migrare i dati nel cloud AWS, tra istanze locali (attraverso la configurazione di un cloud AWS) oppure tra combinazioni di configurazioni locali e cloud.
- AWS SCT - AWS Schema Conversion Tool (AWS SCT) viene utilizzato per convertire lo schema del database da un motore di database a un altro. Il codice personalizzato convertito dallo strumento include viste, procedure memorizzate e funzioni. Qualsiasi codice che lo strumento non è in grado di convertire automaticamente è contrassegnato in modo chiaro in modo che sia possibile convertirlo autonomamente.

Epiche

Pianifica la migrazione

Operazione	Description	Competenze richieste
Convalida la versione e il motore del database di origine e di destinazione.		DBA
Identifica i requisiti hardware per l'istanza del server di destinazione.		DBA, SysAdmin
Identifica i requisiti di archiviazione (tipo e capacità di archiviazione).		DBA, SysAdmin

Operazione	Description	Competenze richieste
Scegli il tipo di istanza corretto in base alla capacità, alle funzionalità di archiviazione e alle funzionalità di rete.		DBA, SysAdmin
Identifica i requisiti di sicurezza dell'accesso alla rete per i database di origine e di destinazione.		DBA, SysAdmin
Identifica la strategia di migrazione delle applicazioni.		DBA SysAdmin, proprietario dell'app

Configura l'infrastruttura

Operazione	Description	Competenze richieste
Crea un cloud privato virtuale (VPC) e sottoreti.		SysAdmin
Crea i gruppi di sicurezza e gli elenchi di controllo degli accessi alla rete (). ACLs		SysAdmin
Configura e avvia un'istanza database Amazon RDS.		DBA, SysAdmin

Migrazione dei dati

Operazione	Description	Competenze richieste
Esegui la migrazione dello schema del database utilizzan do AWS SCT.		DBA

Operazione	Description	Competenze richieste
Migra i dati utilizzando AWS DMS.		DBA

Migrare l'applicazione

Operazione	Description	Competenze richieste
Usa AWS SCT per analizzare e convertire il codice SQL all'interno del codice dell'applicazione.	Per ulteriori informazioni, consulta https://docs.aws.amazon.com/SchemaConversionTool/ latest/userguide/ CHAP _Converting.app.html.	Proprietario dell'app
Segui la strategia di migrazione delle applicazioni.		DBA SysAdmin, proprietario dell'app

Tagliare

Operazione	Description	Competenze richieste
Trasferisci i client applicativi alla nuova infrastruttura.		DBA SysAdmin, proprietario dell'app

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse AWS temporanee.		DBA, SysAdmin
Rivedi e convalida i documenti del progetto.		DBA, SysAdmin

Operazione	Description	Competenze richieste
Raccogli le metriche in tempo utile per la migrazione, percentuale di risorse manuali rispetto a quelle relative agli strumenti, risparmi sui costi, ecc.		DBA, SysAdmin
Chiudi il progetto e fornisci feedback.		

Risorse correlate

Riferimenti

- Documentazione AWS DMS
- Documentazione AWS SCT
- Prezzi di Amazon RDS

Tutorial e video

- Guida introduttiva ad AWS DMS
- Nozioni di base su Amazon RDS
- AWS DMS (video)
- Amazon RDS (video)

Esegui la migrazione di un database Oracle locale ad Amazon RDS for PostgreSQL utilizzando un bystander Oracle e AWS DMS

Cady Motyka, Amazon Web Services

Riepilogo

Questo modello descrive come migrare un database Oracle locale verso uno dei seguenti servizi di database AWS compatibili con PostgreSQL con tempi di inattività minimi:

- Amazon Relational Database Service (Amazon RDS) per PostgreSQL
- Amazon Aurora PostgreSQL-Compatible Edition

La soluzione utilizza AWS Database Migration Service (AWS DMS) per migrare i dati, AWS Schema Conversion Tool (AWS SCT) per convertire lo schema del database e un database Oracle bystander per aiutare a gestire la migrazione. In questa implementazione, il tempo di inattività è limitato al tempo necessario per creare o convalidare tutte le chiavi esterne del database.

La soluzione utilizza anche istanze Amazon Elastic Compute Cloud (Amazon EC2) con un database Oracle bystander per aiutare a controllare il flusso di dati tramite AWS DMS. Puoi sospendere temporaneamente la replica in streaming dal database Oracle locale al bystander Oracle per attivare AWS DMS per recuperare il ritardo sulla convalida dei dati o per utilizzare un altro strumento di convalida dei dati. L'istanza DB Amazon RDS for PostgreSQL o l'istanza DB Aurora compatibile con PostgreSQL e il database bystander avranno gli stessi dati al termine della migrazione delle modifiche correnti da parte di AWS DMS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database Oracle di origine in un data center locale con un database di standby Active Data Guard configurato
- AWS Direct Connect configurato tra il data center locale e AWS Secrets Manager per l'archiviazione dei segreti del database
- Driver Java Database Connectivity (JDBC) per connettori AWS SCT, installati su un computer locale o sull'istanza EC2 in cui è installato AWS SCT

- Familiarità con l'utilizzo di un database Oracle come fonte per AWS DMS
- Familiarità con l'uso di un database PostgreSQL come destinazione per AWS DMS

Limitazioni

- Limite di dimensione del database: 64 TB

Versioni del prodotto

- AWS DMS supporta tutte le edizioni del database Oracle per le versioni 10.2 e successive (per le versioni 10.x), 11g e fino a 12.2, 18c e 19c. Per l'elenco più recente delle versioni supportate, consulta Using an Oracle Database as a Source for AWS DMS. Ti consigliamo di utilizzare la versione più recente di AWS DMS per il supporto più completo della versione e delle funzionalità. Per informazioni sulle versioni dei database Oracle supportate da AWS SCT, consulta la documentazione di AWS SCT.
- AWS DMS supporta le versioni 9.4 e successive di PostgreSQL (per le versioni 9.x), 10.x, 11.x, 12.x e 13.x. Per le informazioni più recenti, consulta Using a PostgreSQL Database as a Target for AWS DMS nella documentazione AWS.

Architecture

Stack tecnologico di origine

- Un database Oracle locale
- Un'EC2 istanza che contiene un astante per il database Oracle

Stack tecnologico Target

- Amazon RDS per PostgreSQL o istanza PostgreSQL Aurora, PostgreSQL 9.3 e versioni successive

Architettura Target

Il diagramma seguente mostra un esempio di flusso di lavoro per la migrazione di un database Oracle a un database AWS compatibile con PostgreSQL utilizzando AWS DMS e un bystander Oracle:

Tools (Strumenti)

- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali.
- AWS Schema Conversion Tool (AWS SCT) supporta migrazioni di database eterogenei convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS.

Epiche

Convertire lo schema del database Oracle in PostgreSQL

Operazione	Description	Competenze richieste
Configura AWS SCT.	Crea un nuovo report e connettiti a Oracle come origine e PostgreSQL come destinazione. In Impostazioni del progetto, vai alla scheda SQL Scripting. Cambia lo script SQL di destinazione in più file. Questi file verranno utilizzati in seguito e denominati come segue: • create_database.sql • create_sequence.sql • create_table.sql • create_view.sql • create_function.sql	DBA
Convertire lo schema del database Oracle.	Nella scheda Azione, scegli Genera rapporto. Quindi,	DBA

Operazione	Description	Competenze richieste
	scegli Converti schema e scegli Salva come SQL.	
Modifica gli script.	Ad esempio, potresti voler modificare lo script se un numero nello schema di origine è stato convertit o in formato numerico in PostgreSQL, ma desideri invece utilizzare BIGINT per prestazioni migliori.	DBA

Crea e configura l'istanza database Amazon RDS

Operazione	Description	Competenze richieste
Crea l'istanza database Amazon RDS.	Nella regione AWS corretta, crea una nuova istanza DB PostgreSQL. Per ulteriori informazioni, consulta Creazione di un'istanza DB PostgreSQL e connessione a un database su un'istanza DB PostgreSQL nella documenta zione di Amazon RDS.	AWS SysAdmin, DBA
Configura le specifiche dell'ista nza DB.	Specificate la versione del motore DB, la classe dell'ista nza DB, l'implementazione Multi-AZ, il tipo di storage e lo storage allocato. Immettere l'identificatore dell'istanza DB, un nome utente principale e una password principale.	AWS SysAdmin, DBA

Operazione	Description	Competenze richieste
Configura rete e sicurezza.	Specificare il cloud privato virtuale (VPC), il gruppo di sottoreti, l'accessibilità pubblica, la preferenza della zona di disponibilità e i gruppi di sicurezza.	DBA, SysAdmin
Configurare le opzioni del database.	Specificare il nome del database, la porta, il gruppo di parametri, la crittografia e la chiave KMS.	AWS SysAdmin, DBA
Configurare i backup.	Specificate il periodo di conservazione del backup, la finestra di backup, l'ora di inizio, la durata e se copiare i tag nelle istantanee.	AWS SysAdmin, DBA
Configura le opzioni di monitoraggio.	Attiva o disattiva il monitorag gio avanzato e gli approfond imenti sulle prestazioni.	AWS SysAdmin, DBA
Configura le opzioni di manutenzione.	Specificare l'aggiornamento automatico della versione secondaria, la finestra di manutenzione e il giorno, l'ora e la durata di inizio.	AWS SysAdmin, DBA

Operazione	Description	Competenze richieste
Esegui gli script di pre-migrazione da AWS SCT.	Sull'istanza Amazon RDS, esegui i seguenti script generati da AWS SCT: • create_database.sql • create_sequence.sql • create_table.sql • create_view.sql • create_function.sql	AWS SysAdmin, DBA

Configura il bystander Oracle in Amazon EC2

Operazione	Description	Competenze richieste
Configura la rete per Amazon EC2.	Crea il nuovo VPC, le sottoreti, il gateway Internet, le tabelle di routing e i gruppi di sicurezza.	AWS SysAdmin
Crea l' EC2 istanza.	Nella regione AWS appropriata, crea una nuova EC2 istanza. Seleziona Amazon Machine Image (AMI), scegli la dimensione dell'istanza e configura i dettagli dell'istanza: numero di istanze (1), VPC e sottorete creati nell'attività precedente, assegnazione automatica dell'IP pubblico e altre opzioni. Aggiungi storage, configura i gruppi di sicurezza e avvia. Quando richiesto, create e salvate	AWS SysAdmin

Operazione	Description	Competenze richieste
	una key pair per il passaggio successivo.	
Connect il database di origine Oracle all' EC2 istanza.	Copia l'indirizzo IP IPv4 pubblico e il DNS in un file di testo e connettiti utilizzando SSH nel modo seguente: ssh -i «your_file.pem» EC2-user@ <your-IP - -DNS>. address-or-public	AWS SysAdmin
Configura l'host iniziale per un passante in Amazon. EC2	Configura chiavi SSH, profilo bash, ORATAB e link simbolici . Crea directory Oracle.	AWS SysAdmin, amministratore Linux
Configura la copia del database per un passante in Amazon EC2	Usa RMAN per creare una copia del database, abilitare la registrazione supplementare e creare il file di controllo in standby. Al termine della copia, posizionate il database in modalità di ripristino.	AWS SysAdmin, DBA

Operazione	Description	Competenze richieste
Configura Oracle Data Guard.	Modifica il file listener.ora e avvia il listener. Imposta una nuova destinazione di archiviazione. Metti lo spettatore in modalità di ripristino, sostituisci i file temporanei per evitare future danneggiamenti, installa un crontab se necessario per evitare che la directory di archivio si esaurisca lo spazio e modifica il manage-trclog-files-oraclefile.cfg come sorgente e standby.	AWS SysAdmin, DBA
Prepara il database Oracle per sincronizzare la spedizione.	Aggiungi i file di registro in standby e modifica la modalità di ripristino. Modifica il log shipping in SYNC AFFIRM sia sulla sorgente primaria che sulla sorgente standby. Passa ai log primari, conferma tramite l'alert log di Amazon EC2 bystander che stai utilizzando i file di log di standby e conferma che il redo stream scorra in SYNC.	AWS SysAdmin, DBA

Migrazione dei dati con AWS DMS

Operazione	Description	Competenze richieste
Crea un'istanza di replica in AWS DMS.	Completa i campi per il nome, la classe dell'istanza,	AWS SysAdmin, DBA

Operazione	Description	Competenze richieste
	il VPC (come l' EC2 istanza Amazon), Multi-AZ e l'accessi bilità pubblica. In Advance, specifica lo storage allocato, il gruppo di sottoreti, la zona di disponibilità, i gruppi di sicurezza VPC e la chiave AWS Key Management Service (AWS KMS).	
Crea l'endpoint del database di origine.	Specificare il nome dell'endp oint, il tipo, il motore di origine (Oracle), il nome del server (nome DNS EC2 privato Amazon), la porta, la modalità SSL, il nome utente, la password, il SID, il VPC (specifica il VPC che ha l'istanza di replica) e l'istanza di replica. Per testare la connessione, scegli Run Test, quindi crea l'endpoint. Puoi anche configurare le seguenti impostazioni avanzate: maxFileSizee numberDat aTypeScale.	AWS SysAdmin, DBA
Connetti AWS DMS ad Amazon RDS per PostgreSQ L.	Crea un gruppo di sicurezza di migrazione per le connessioni tra. VPCs	AWS SysAdmin, DBA

Operazione	Description	Competenze richieste
Crea l'endpoint del database di destinazione.	Specificare il nome dell'endpoint, il tipo, il motore di origine (PostgreSQL), il nome del server (endpoint Amazon RDS), la porta, la modalità SSL, il nome utente, la password, il nome del database, il VPC (specifica il VPC che contiene l'istanza di replica) e l'istanza di replica. Per testare la connessione, scegli Esegui test, quindi crea l'endpoint. Puoi anche configurare le seguenti impostazioni avanzate: maxFileSize e numberDataTypeScale.	AWS SysAdmin, DBA
Crea l'attività di replica AWS DMS.	Specificare il nome dell'attività, l'istanza di replica, gli endpoint di origine e di destinazione e l'istanza di replica. Per il tipo di migrazione, scegli Migra i dati esistenti e replica le modifiche in corso. Deselezionare la casella di controllo Avvia attività alla creazione.	AWS SysAdmin, DBA

Operazione	Description	Competenze richieste
Configura le impostazioni delle attività di replica di AWS DMS.	Per la modalità di preparazione della tabella di destinazione, scegli Non fare nulla. Interrompi l'operazione dopo il completamento del caricamento completo (per creare le chiavi primarie). Specificate la modalità LOB limitata o completa e attivate le tabelle di controllo. Facoltativamente, è possibile configurare l'impostazione CommitRateavanzata.	DBA
Configura le mappature delle tabelle.	Nella sezione Mappature delle tabelle, crea una regola di inclusione per tutte le tabelle in tutti gli schemi inclusi nella migrazione, quindi crea una regola di esclusione. Aggiungi tre regole di trasformazione per convertire i nomi di schema, tabella e colonna in lettere minuscole e aggiungi tutte le altre regole necessarie per questa migrazione specifica.	DBA
Avvia l’attività .	Avvia l'attività di replica. Accertatevi che il carico sia in esecuzione a pieno carico. Esegui ALTER SYSTEM SWITCH LOGFILE sul database Oracle primario per avviare l'attività.	DBA

Operazione	Description	Competenze richieste
Esegui gli script di migrazione intermedia da AWS SCT.	In Amazon RDS for PostgreSQL, esegui i seguenti script generati da AWS SCT: • create_index.sql • create_constraint.sql	DBA
Riavviare l'attività per continuare l'acquisizione dei dati delle modifiche (CDC).	Esegui VACUUM sull'istanza DB Amazon RDS for PostgreSQL e riavvia l'attività AWS DMS per applicare le modifiche CDC memorizzate nella cache.	DBA

Passa al database PostgreSQL

Operazione	Description	Competenze richieste
Controlla i log e le tabelle di convalida di AWS DMS per eventuali errori.	Controlla e correggi eventuali errori di replica o convalida.	DBA
Interrompi tutte le dipendenze Oracle.	Arresta tutte le dipendenze Oracle, spegni i listener sul database Oracle ed esegui ALTER SYSTEM SWITCH LOGFILE. Interrompi l'attività AWS DMS quando non mostra alcuna attività.	DBA
Esegui gli script post-migrazione da AWS SCT.	In Amazon RDS for PostgreSQL, esegui i seguenti script generati da AWS SCT:	DBA

Operazione	Description	Competenze richieste
	• create_foreign_key_constraint.sql • create_triggers.sql	
Completa i passaggi aggiuntivi di Amazon RDS for PostgreSQL.	Incrementa le sequenze in modo che corrispondano a quelle di Oracle, se necessario, esegui VACUUM e ANALYZE e scatta un'istantanea per garantire la conformità.	DBA
Apri le connessioni ad Amazon RDS for PostgreSQL.	Rimuovi i gruppi di sicurezza AWS DMS da Amazon RDS for PostgreSQL, aggiungi gruppi di sicurezza di produzione e indirizza le tue applicazioni verso il nuovo database.	DBA
Pulisci gli oggetti AWS DMS.	Rimuovi gli endpoint, le attività di replica, le istanze di replica e l'istanza. EC2	SysAdmin, DBA

Risorse correlate

- Documentazione AWS DMS
- Documentazione AWS SCT
- Prezzi di Amazon RDS per PostgreSQL

Esegui la migrazione di un database Oracle ad Amazon Redshift utilizzando AWS DMS e AWS SCT

Piyush Goyal e Brian Motzer, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni per la migrazione dei database Oracle a un data warehouse cloud Amazon Redshift nel cloud Amazon Web Services (AWS) utilizzando AWS Database Migration Service (AWS DMS) e AWS Schema Conversion Tool (AWS SCT). Il modello copre i database Oracle di origine che sono locali o installati su un'istanza Amazon Elastic Compute Cloud (Amazon EC2). Copre anche Amazon Relational Database Service (Amazon RDS) per database Oracle.

Prerequisiti e limitazioni

Prerequisiti

- Un database Oracle in esecuzione in un data center locale o nel cloud AWS
- Un account AWS attivo
- Familiarità con l'utilizzo di un database Oracle come fonte per AWS DMS
- Familiarità con l'uso di un database Amazon Redshift come destinazione per AWS DMS
- Conoscenza di Amazon RDS, Amazon Redshift, le tecnologie di database applicabili e SQL
- Driver Java Database Connectivity (JDBC) per connettori AWS SCT, su cui è installato AWS SCT

Versioni del prodotto

- Per i database Oracle autogestiti, AWS DMS supporta tutte le edizioni dei database Oracle per le versioni 10.2 e successive (per le versioni 10). x), 11g e fino a 12.2, 18c e 19c. Per i database Amazon RDS for Oracle gestiti da AWS, AWS DMS supporta tutte le edizioni dei database Oracle per le versioni 11g (versioni 11.2.0.4 e successive) e fino a 12.2, 18c e 19c. Ti consigliamo di utilizzare la versione più recente di AWS DMS per il supporto più completo della versione e delle funzionalità.

Architecture

Stack tecnologico di origine

Una delle seguenti:

- Un database Oracle locale
- Un database Oracle su un'istanza EC2
- Un'istanza DB Amazon RDS per Oracle

Stack tecnologico Target

- Amazon Redshift

Architettura di destinazione

Da un database Oracle in esecuzione nel cloud AWS ad Amazon Redshift:

Da un database Oracle in esecuzione in un data center locale ad Amazon Redshift:

Tools (Strumenti)

- AWS DMS - AWS Data Migration Service (AWS DMS) ti aiuta a migrare i database su AWS in modo rapido e sicuro. Il database di origine rimane pienamente operativo durante la migrazione, riducendo al minimo i tempi di inattività delle applicazioni che si basano sul database. AWS DMS può migrare i dati da e verso i database commerciali e open source più utilizzati.
- AWS SCT - AWS Schema Conversion Tool (AWS SCT) può essere utilizzato per convertire lo schema di database esistente da un motore di database a un altro. Supporta vari motori di database, tra cui Oracle, SQL Server e PostgressSQL, come sorgenti.

Epiche

Preparati per la migrazione

Operazione	Description	Competenze richieste
Convalida le versioni del database.	Convalida le versioni del database di origine e di destinazione e assicurat i che siano supportate da AWS DMS. Per informazioni	DBA

Operazione	Description	Competenze richieste
	sulle versioni supportate di Oracle Database, consulta Using an Oracle database as a source for AWS DMS. Per informazioni sull'utilizzo di Amazon Redshift come destinazione, consulta Usare un database Amazon Redshift come destinazione per AWS DMS.	
Crea un VPC e un gruppo di sicurezza.	Nel tuo account AWS, crea un cloud privato virtuale (VPC), se non esiste. Crea un gruppo di sicurezza per il traffico in uscita verso i database di origine e destinazione. Per ulteriori informazioni, consulta la documentazione di Amazon Virtual Private Cloud (Amazon VPC).	Amministratore di sistema
Installa AWS SCT.	Scarica e installa la versione più recente di AWS SCT e i driver corrispondenti. Per ulteriori informazioni, consulta Installazione, verifica e aggiornamento di AWS SCT.	DBA
Crea un utente per il task AWS DMS.	Crea un utente AWS DMS nel database di origine e concedigli i privilegi READ. Questo utente verrà utilizzato sia da AWS SCT che da AWS DMS.	DBA

Operazione	Description	Competenze richieste
Verifica la connettività DB.	Verifica la connettività all'istanza DB di Oracle.	DBA
Crea un nuovo progetto in AWS SCT.	Apri lo strumento AWS SCT e crea un nuovo progetto.	DBA
Analizza lo schema Oracle da migrare.	Usa AWS SCT per analizzare lo schema da migrare e generare un rapporto di valutazione della migrazione e del database. Per ulteriori informazioni, consulta Creazione di un report di valutazione della migrazione del database nella documentazione di AWS SCT.	DBA
Esamina il rapporto di valutazione.	Esamina il rapporto per verificare la fattibilità della migrazione. Alcuni oggetti DB potrebbero richiedere la conversione manuale. Per ulteriori informazioni sul report, consulta Visualizzazione del rapporto di valutazione nella documentazione di AWS SCT.	DBA

Preparare il database di destinazione

Operazione	Description	Competenze richieste
Crea un cluster Amazon Redshift.	Crea un cluster Amazon Redshift all'interno del VPC creato in precedenza. Per ulteriori informazioni, consulta	DBA

Operazione	Description	Competenze richieste
	i cluster Amazon Redshift nella documentazione di Amazon Redshift.	
Crea utenti del database.	Estrai l'elenco di utenti, ruoli e concessioni dal database di origine Oracle. Crea utenti nel database Amazon Redshift di destinazione e applica i ruoli del passaggio precedente.	DBA
Valuta i parametri del database.	Esamina le opzioni del database, i parametri, i file di rete e i link al database dal database di origine Oracle e valuta la loro applicabilità alla destinazione.	DBA
Applica tutte le impostazioni pertinenti all'obiettivo.	Per ulteriori informazioni su questo passaggio, consulta il riferimento alla configurazione nella documentazione di Amazon Redshift.	DBA

Crea oggetti nel database di destinazione

Operazione	Description	Competenze richieste
Crea un utente AWS DMS nel database di destinazione.	Crea un utente AWS DMS nel database di destinazione e concedigli i privilegi di lettura e scrittura. Convalida la connettività da AWS SCT.	DBA

Operazione	Description	Competenze richieste
Converti lo schema, esamina il report SQL e salva eventuali errori o avvisi.	Per ulteriori informazioni, consulta Conversione degli schemi di database utilizzando AWS SCT nella documentazione di AWS SCT.	DBA
Applica le modifiche allo schema al database di destinazione o salvale come file.sql.	Per istruzioni, consulta Salvare e applicare lo schema convertito in AWS SCT nella documentazione di AWS SCT.	DBA
Convalida gli oggetti nel database di destinazione.	Convalida gli oggetti creati nel passaggio precedente nel database di destinazione. Riscrivi o riprogetta gli oggetti che non sono stati convertiti correttamente.	DBA
Disabilita le chiavi esterne e i trigger.	Disabilita qualsiasi chiave esterna e trigger. Questi possono causare problemi di caricamento dei dati durante il processo di caricamento completo durante l'esecuzione di AWS DMS.	DBA

Migrazione dei dati con AWS DMS

Operazione	Description	Competenze richieste
Creare un'istanza di replica di AWS DMS.	Accedi alla Console di gestione AWS e apri la console AWS DMS. Nel pannello di navigazione, scegli Istanze di replica,	DBA

Operazione	Description	Competenze richieste
	Crea istanza di replica. Per istruzioni dettagliate, consulta il passaggio 1 in Getting started with AWS DMS nella documentazione di AWS DMS.	
Crea endpoint di origine e destinazione.	Crea endpoint di origine e di destinazione, verifica la connessione dall'istanza di replica agli endpoint di origine e di destinazione. Per istruzioni dettagliate, consulta il passaggio 2 in Getting started with AWS DMS nella documentazione di AWS DMS.	DBA
Creare un'attività di replica.	Creare un'attività di replica e selezionare il metodo di migrazione appropriato. Per istruzioni dettagliate, consulta il passaggio 3 in Getting started with AWS DMS nella documentazione di AWS DMS.	DBA
Avvia la replica dei dati.	Avvia l'attività di replica e monitora i log per eventuali errori.	DBA

Migra la tua applicazione

Operazione	Description	Competenze richieste
Crea server di applicazioni.	Crea i nuovi server delle applicazioni su AWS.	Proprietario dell'applicazione
Esegui la migrazione del codice dell'applicazione.	Migrare il codice dell'appl icazione sui nuovi server.	Proprietario dell'applicazione
Configurare il server delle applicazioni.	Configura il server delle applicazioni per il database e i driver di destinazione.	Proprietario dell'applicazione
Ottimizza il codice dell'appl icazione.	Ottimizza il codice dell'appl icazione per il motore di destinazione.	Proprietario dell'applicazione

Passa al database di destinazione

Operazione	Description	Competenze richieste
Convalida gli utenti.	Nel database Amazon Redshift di destinazione, convalida gli utenti e concedi loro ruoli e privilegi.	DBA
Verifica che l'applicazione sia bloccata.	Assicuratevi che l'applica zione sia bloccata, per evitare ulteriori modifiche.	Proprietario dell'applicazione
Convalida i dati.	Convalida i dati nel database Amazon Redshift di destinazi one.	DBA
Abilita chiavi esterne e trigger.	Abilita chiavi esterne e trigger nel database Amazon Redshift di destinazione.	DBA

Operazione	Description	Competenze richieste
Connect al nuovo database.	Configura l'applicazione per la connessione al nuovo database Amazon Redshift.	Proprietario dell'applicazione
Eseguire i controlli finali.	Esegui un controllo finale e completo del sistema prima di andare in diretta.	DBA, proprietario dell'applicazione
Trasmetti in diretta.	Trasmetti online il database Amazon Redshift di destinazione.	DBA

Chiudi il progetto di migrazione

Operazione	Description	Competenze richieste
Chiudi le risorse AWS temporanee.	Chiudi le risorse AWS temporanee come l'istanza di replica AWS DMS e l' EC2 istanza utilizzata per AWS SCT.	DBA, amministratore di sistema
Rivedi i documenti.	Rivedi e convalida i documenti del progetto di migrazione.	DBA, amministratore di sistema
Raccogli le metriche.	Raccogli informazioni sul progetto di migrazione, ad esempio il tempo necessario per la migrazione, la percentuale di attività manuali rispetto a quelle eseguite con l'ausilio di strumenti e il risparmio totale sui costi.	DBA, amministratore di sistema

Operazione	Description	Competenze richieste
Chiudi il progetto.	Chiudi il progetto e fornisci feedback.	DBA, amministratore di sistema

Risorse correlate

Riferimenti

- Guida per l'utente di AWS DMS
- Guida per l'utente di AWS SCT
- Guida introduttiva ad Amazon Redshift

Tutorial e video

- Scopri di più su AWS SCT e AWS DMS (presentazione da AWS re:Invent 2019)
- Guida introduttiva ad AWS Database Migration Service

Esegui la migrazione di un database Oracle ad Aurora PostgreSQL utilizzando AWS DMS e AWS SCT

Senthil Ramasamy, Amazon Web Services

Riepilogo

Questo modello descrive come migrare un database Oracle verso l'edizione compatibile con Amazon Aurora PostgreSQL utilizzando AWS Data Migration Service (AWS DMS) e AWS Schema Conversion Tool (AWS SCT).

Il modello copre i database Oracle di origine locali, i database Oracle installati su istanze Amazon Elastic Compute Cloud (Amazon EC2) e Amazon Relational Database Service (Amazon RDS) per i database Oracle. Il pattern converte questi database in Aurora PostgreSQL compatibili.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un database Oracle in un data center locale o nel cloud AWS.
- Client SQL installati su un computer locale o su un' EC2 istanza.
- Driver Java Database Connectivity (JDBC) per connettori AWS SCT, installati su un computer locale o su un'istanza EC2 in cui è installato AWS SCT.

Limitazioni

- Limite di dimensione del database: 128 TB
- Se il database di origine supporta un'applicazione commerciale off-the-shelf (COTS) o è specifico del fornitore, potrebbe non essere possibile convertirlo in un altro motore di database. Prima di utilizzare questo pattern, verifica che l'applicazione supporti la compatibilità con Aurora PostgreSQL.

Versioni del prodotto

- Per i database Oracle autogestiti, AWS DMS supporta tutte le edizioni dei database Oracle per le versioni 10.2 e successive (per le versioni 10.x), 11g e fino a 12.2, 18c e 19c. Per l'elenco più recente delle versioni di database Oracle supportate (sia autogestite che Amazon RDS for Oracle),

consulta Utilizzo di un database Oracle come origine per AWS DMS e Utilizzo di un database PostgreSQL come destinazione per AWS DMS.

- Ti consigliamo di utilizzare la versione più recente di AWS DMS per il supporto più completo della versione e delle funzionalità. Per informazioni sulle versioni dei database Oracle supportate da AWS SCT, consulta la documentazione di AWS SCT.
- Aurora supporta le versioni di PostgreSQL elencate nelle versioni di Amazon Aurora PostgreSQL e nelle versioni del motore.

Architecture

Stack tecnologico di origine

Una delle seguenti:

- Un database Oracle locale
- Un database Oracle su un'istanza EC2
- Un'istanza DB Amazon RDS per Oracle

Stack tecnologico Target

- Compatibile con Aurora PostgreSQL

Architettura Target

Architettura di migrazione dei dati

- Da un database Oracle in esecuzione nel cloud AWS

- Da un database Oracle in esecuzione in un data center locale

Tools (Strumenti)

- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali.

- AWS Schema Conversion Tool (AWS SCT) supporta migrazioni di database eterogenei convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione.

Epiche

Preparati per la migrazione

Operazione	Description	Competenze richieste
Preparare il database di origine.	Per preparare il database di origine, consulta Using Oracle Database as a source for AWS SCT nella documenta zione di AWS SCT.	DBA
Crea un' EC2 istanza per AWS SCT.	Crea e configura un' EC2 istanza per AWS SCT, se necessario.	DBA
Scarica AWS SCT.	Scarica la versione più recente di AWS SCT e i driver associati. Per ulteriori informazioni, consulta Installaz ione, verifica e aggiornamento di AWS SCT nella documenta zione di AWS SCT.	DBA
Aggiungi utenti e autorizza zioni.	Aggiungi e convalida gli utenti e le autorizzazioni prerequisiti nel database di origine.	DBA
Crea un progetto AWS SCT.	Crea un progetto AWS SCT per il carico di lavoro e connettiti al database di origine. Per istruzioni, consulta Creazione di un progetto AWS SCT e Aggiungere server di	DBA

Operazione	Description	Competenze richieste
	database nella documenta zione di AWS SCT.	
Valuta la fattibilità.	Genera un rapporto di valutazione, che riepiloga le azioni da intraprendere per gli schemi che non possono essere convertiti automatic amente e fornisce stime degli sforzi di conversione manuali. Per ulteriori informazioni, consulta Creazione e revisione del report di valutazione della migrazione del database nella documentazione di AWS SCT.	DBA

Preparare il database di destinazione

Operazione	Description	Competenze richieste
Crea un'istanza database Amazon RDS di destinazione.	Crea un'istanza database Amazon RDS di destinazione utilizzando Amazon Aurora come motore di database. Per istruzioni, consulta Creazione di un'istanza database Amazon RDS nella documenta zione di Amazon RDS.	DBA
Estrai utenti, ruoli e autorizza zioni.	Estrai l'elenco di utenti, ruoli e autorizzazioni dal database di origine.	DBA

Operazione	Description	Competenze richieste
Mappa gli utenti.	Mappare gli utenti esistenti del database ai nuovi utenti del database.	Proprietario dell'app
Creare utenti.	Crea utenti nel database di destinazione.	DBA, proprietario dell'app
Applica ruoli.	Applica i ruoli del passaggio precedente al database di destinazione.	DBA
Controlla opzioni, parametri , file di rete e collegamenti al database.	Esamina le opzioni, i parametri , i file di rete e i collegamenti al database di origine, quindi valuta la loro applicabilità al database di destinazione.	DBA
Applica le impostazioni.	Applica tutte le impostazi oni pertinenti al database di destinazione.	DBA

Trasferisci oggetti

Operazione	Description	Competenze richieste
Configura la connettività AWS SCT.	Configura la connettività AWS SCT al database di destinazi one.	DBA
Converti lo schema utilizzando AWS SCT.	AWS SCT converte automatic amente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database	DBA

Operazione	Description	Competenze richieste
	di destinazione. Qualsiasi codice che lo strumento non è in grado di convertire automaticamente è chiaramente contrassegnato in modo da poterlo convertire manualmente.	
Esamina il rapporto.	Esamina il report SQL generato e salva eventuali errori e avvisi.	DBA
Applica modifiche automatiche allo schema.	Applica modifiche automatiche allo schema al database di destinazione o salvale come file.sql.	DBA
Convalida gli oggetti.	Verifica che AWS SCT abbia creato gli oggetti sulla destinazione.	DBA
Gestisci gli elementi che non sono stati convertiti.	Riscrivi, rifiuta o riprogetta manualmente gli elementi che non sono stati convertiti automaticamente.	DBA, proprietario dell'app
Applica le autorizzazioni per ruoli e utenti.	Applica il ruolo generato e le autorizzazioni utente ed esamina le eventuali eccezioni.	DBA

Migrare i dati

Operazione	Description	Competenze richieste
Determina il metodo.	Determina il metodo per la migrazione dei dati.	DBA
Crea un'istanza di replica.	Crea un'istanza di replica dalla console AWS DMS. Per ulteriori informazioni, consulta Lavorare con un'istanza di replica AWS DMS nella documentazione di AWS DMS.	DBA
Crea gli endpoint di origine e di destinazione.	Per creare endpoint, segui le istruzioni in Creazione di endpoint di origine e destinazione nella documentazione di AWS DMS.	DBA
Creare un'attività di replica.	Per creare un'attività, consulta Working with AWS DMS tasks nella documentazione di AWS DMS.	DBA
Avvia l'attività di replica e monitora i log.	Per ulteriori informazioni su questo passaggio, consulta Monitoraggio delle attività di AWS DMS nella documentazione di AWS DMS.	DBA

Migrare l'applicazione

Operazione	Description	Competenze richieste
Analizza e converti gli elementi SQL nel codice dell'applicazione.	Usa AWS SCT per analizzare e convertire gli elementi SQL nel codice dell'applicazione. Quando converti lo schema del database da un motore a un altro, è anche necessario aggiornare il codice SQL nelle applicazioni, per interagire con il nuovo motore di database al posto di quello precedente. Puoi visualizzare, analizzare, modificare e salvare il codice SQL convertito.	Proprietario dell'app
Crea server di applicazioni.	Crea i nuovi server delle applicazioni su AWS.	Proprietario dell'app
Esegui la migrazione del codice dell'applicazione.	Migrare il codice dell'appl icazione sui nuovi server.	Proprietario dell'app
Configura i server delle applicazioni.	Configura i server delle applicazioni per il database e i driver di destinazione.	Proprietario dell'app
Correggi il codice.	Correggi qualsiasi codice specifico del motore di database di origine dell'appl icazione.	Proprietario dell'app
Ottimizza il codice.	Ottimizza il codice dell'appl icazione per il motore di database di destinazione.	Proprietario dell'app

Tagliare

Operazione	Description	Competenze richieste
Trasferiscilo al database di destinazione.	Esegui il cutover sul nuovo database.	DBA
Blocca l'applicazione.	Blocca l'applicazione da eventuali ulteriori modifiche.	Proprietario dell'app
Convalida le modifiche.	Verifica che tutte le modifiche siano state propagate al database di destinazione.	DBA
Reindirizzamento al database di destinazione.	Indirizza i nuovi server delle applicazioni verso il database di destinazione.	Proprietario dell'app
Controllate tutto.	Esegui un controllo finale e completo del sistema.	Proprietario dell'app
Trasmetti in diretta.	Completa le attività finali di cutover.	Proprietario dell'app

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse temporanee.	Chiudi le risorse AWS temporanee come l'istanza di replica AWS DMS e l' EC2 istanza utilizzata per AWS SCT.	DBA, proprietario dell'app
Aggiorna il feedback.	Aggiorna il feedback sul processo AWS DMS per i team interni.	DBA, proprietario dell'app

Operazione	Description	Competenze richieste
Modifica il processo e i modelli.	Rivedi il processo AWS DMS e, se necessario, migliora il modello.	DBA, proprietario dell'app
Convalida i documenti.	Rivedi e convalida i documenti del progetto.	DBA, proprietario dell'app
Raccogli le metriche.	Raccogli le metriche per valutare i tempi di migrazione, la percentuale di risparmio sui costi manuali rispetto a quelli degli strumenti e così via.	DBA, proprietario dell'app
Chiudi il progetto.	Chiudi il progetto di migrazione e fornisci feedback alle parti interessate.	DBA, proprietario dell'app

Risorse correlate

Riferimenti

- Utilizzo di un database Oracle come sorgente per AWS DMS
- Utilizzo di un database PostgreSQL come destinazione per AWS Database Migration Service
- Playbook di migrazione da Oracle Database 11g/12c ad Amazon Aurora con compatibilità PostgreSQL (9.6.x)
- Playbook sulla migrazione da Oracle Database 19c ad Amazon Aurora con compatibilità PostgreSQL (12.4)
- Migrazione di un database Amazon RDS for Oracle verso Amazon Aurora PostgreSQL Compatible Edition
- Servizio di migrazione dei dati AWS
- Strumento di conversione dello schema AWS
- Esegui la migrazione da Oracle ad Amazon Aurora
- Prezzi di Amazon RDS

Tutorial e video

- Procedure dettagliate per la migrazione del database Step-by-Step
- Guida introduttiva ad AWS DMS
- Nozioni di base su Amazon RDS
- AWS Data Migration Service (video)
- Migrazione di un database Oracle a PostgreSQL (video)

Informazioni aggiuntive

.

Esegui la migrazione dei dati da un database Oracle locale ad Aurora PostgreSQL

Michelle Deng e Shunan Xiang, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni per la migrazione dei dati da un database Oracle locale all'edizione compatibile con Amazon Aurora PostgreSQL. Mira a una strategia di migrazione dei dati online con tempi di inattività minimi per database Oracle da più terabyte che contengono tabelle di grandi dimensioni con attività DML (High Data Manipulation Language). Un database di standby Oracle Active Data Guard viene utilizzato come origine per scaricare la migrazione dei dati dal database primario. La replica dal database primario Oracle allo standby può essere sospesa durante il pieno carico per evitare errori ORA-01555.

Le colonne di tabella nelle chiavi primarie (PKs) o nelle chiavi esterne (FKs), con tipo di dati NUMBER, vengono comunemente utilizzate per memorizzare numeri interi in Oracle. Ti consigliamo di convertirli in INT o BIGINT in PostgreSQL per prestazioni migliori. Puoi utilizzare AWS Schema Conversion Tool (AWS SCT) per modificare la mappatura dei tipi di dati predefinita per le colonne PK e FK. (Per ulteriori informazioni, consulta il post AWS sul blog Convertire il tipo di dati NUMBER da Oracle a PostgreSQL.) La migrazione dei dati in questo modello utilizza AWS Database Migration Service (AWS DMS) sia per l'acquisizione dei dati a pieno carico che per quella di modifica (CDC).

Puoi anche utilizzare questo modello per migrare un database Oracle locale su Amazon Relational Database Service (Amazon RDS) per PostgreSQL o un database Oracle ospitato su Amazon Elastic Compute Cloud (Amazon) verso Amazon RDS for PostgreSQL o Aurora PostgreSQL compatibile. EC2

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database di origine Oracle in un data center locale con configurazione in standby Active Data Guard
- AWS Direct Connect configurato tra il data center locale e il cloud AWS
- Familiarità con l'utilizzo di un database Oracle come fonte per AWS DMS
- Familiarità con l'uso di un database PostgreSQL come destinazione per AWS DMS

Limitazioni

- I cluster di database Amazon Aurora possono essere creati con un massimo di 128 TiB di storage. Le istanze di database Amazon RDS for PostgreSQL possono essere create con un massimo di 64 TiB di storage. Per le informazioni più recenti sullo storage, consulta lo storage e l'affidabilità di Amazon Aurora e lo storage di istanze DB Amazon RDS nella documentazione AWS.

Versioni del prodotto

- AWS DMS supporta tutte le edizioni del database Oracle per le versioni 10.2 e successive (per le versioni 10.x), 11g e fino a 12.2, 18c e 19c. Per l'elenco più recente delle versioni supportate, consulta Using an Oracle Database as a Source for AWS DMS nella documentazione AWS.

Architecture

Stack tecnologico di origine

- Database Oracle locali con configurazione in standby Oracle Active Data Guard

Stack tecnologico Target

- Compatibile con Aurora PostgreSQL

Architettura di migrazione dei dati

Tools (Strumenti)

- AWS DMS - AWS Database Migration Service (AWS DMS) supporta diversi database di origine e destinazione. Consulta Using an Oracle Database as a Source for AWS DMS nella documentazione di AWS DMS per un elenco delle versioni ed edizioni del database Oracle di origine e destinazione supportate. Se il database di origine non è supportato da AWS DMS, devi selezionare un altro metodo per la migrazione dei dati nella Fase 6 (nella sezione Epics). Nota importante: poiché si tratta di una migrazione eterogenea, è necessario innanzitutto verificare se il database supporta un'applicazione commerciale (COTS). off-the-shelf Se l'applicazione è COTS, consulta il fornitore per confermare che la compatibilità con Aurora PostgreSQL sia supportata prima di procedere. Per ulteriori informazioni, consulta le procedure dettagliate per la Step-by-Step migrazione di AWS DMS nella documentazione AWS.

- AWS SCT - L'AWS Schema Conversion Tool (AWS SCT) facilita le migrazioni di database eterogenei convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione. Il codice personalizzato convertito dallo strumento include viste, procedure memorizzate e funzioni. Qualsiasi codice che lo strumento non è in grado di convertire automaticamente è contrassegnato in modo chiaro in modo che sia possibile convertirlo autonomamente.

Epiche

Pianifica la migrazione

Operazione	Description	Competenze richieste
Convalida le versioni del database di origine e di destinazione.		DBA
Installa AWS SCT e i driver.		DBA
Aggiungi e convalida gli utenti dei prerequisiti AWS SCT e il database grants-source.		DBA
Crea un progetto AWS SCT per il carico di lavoro e connettiti al database di origine.		DBA
Genera un rapporto di valutazione e valuta la fattibilità.		DBA, proprietario dell'app

Preparare il database di destinazione

Operazione	Description	Competenze richieste
Crea un database di destinazione compatibile con Aurora PostgreSQL.		DBA
Estrai l'elenco degli utenti, dei ruoli e delle concessioni dal database di origine.		DBA
Associa gli utenti esistenti del database ai nuovi utenti del database.		Proprietario dell'app
Crea utenti nel database di destinazione.		DBA
Applica i ruoli del passaggio precedente al database Aurora di destinazione compatibile con PostgreSQL.		DBA
Esamina le opzioni del database, i parametri, i file di rete e i collegamenti al database dal database di origine e valuta la loro applicabilità al database di destinazione.		DBA, proprietario dell'app
Applica tutte le impostazioni pertinenti al database di destinazione.		DBA

Prepararsi per la conversione del codice oggetto del database

Operazione	Description	Competenze richieste
Configura la connettività AWS SCT al database di destinazione.		DBA
Converti lo schema in AWS SCT e salva il codice convertito come file.sql.		DBA, proprietario dell'app
Converti manualmente tutti gli oggetti del database che non sono stati convertiti automaticamente.		DBA, proprietario dell'app
Ottimizza la conversione del codice del database.		DBA, proprietario dell'app
Separa il file.sql in più file.sql in base al tipo di oggetto.		DBA, proprietario dell'app
Convalida gli script SQL nel database di destinazione.		DBA, proprietario dell'app

Preparati per la migrazione dei dati

Operazione	Description	Competenze richieste
Creare un'istanza di replica di AWS DMS.		DBA
Crea gli endpoint di origine e di destinazione.	Se il tipo di dati di PKs and FKs viene convertito da NUMBER in Oracle a BIGINT in PostgreSQL, valuta la possibilità di specificare	DBA

Operazione	Description	Competenze richieste
	l'attributo di connessione `numberDataTypeScale=-2` quando crei l'endpoint di origine.	

Migrazione dei dati: a pieno carico

Operazione	Description	Competenze richieste
Crea lo schema e le tabelle nel database di destinazione.		DBA
Crea attività a pieno carico di AWS DMS raggruppando tabelle o suddividendo una tabella di grandi dimensioni in base alle dimensioni della tabella.		DBA
Arresta le applicazioni sui database Oracle di origine per un breve periodo.		Proprietario dell'app
Verificare che il database di standby Oracle sia sincrono con il database primario e interrompere la replica dal database primario al database di standby.		DBA, proprietario dell'app
Avvia le applicazioni sul database Oracle di origine.		Proprietario dell'app
Avvia le attività di caricamento completo di AWS DMS in parallelo dal database di		DBA

Operazione	Description	Competenze richieste
standby Oracle al database Aurora compatibile con PostgreSQL.		
Crea PKs indici secondari dopo il completamento del caricamento completo.		DBA
Convalida i dati.		DBA

Migrazione dei dati — CDC

Operazione	Description	Competenze richieste
Crea attività di replica continue di AWS DMS specificando un'ora di inizio CDC o un numero di modifica del sistema (SCN) personalizzato quando lo standby di Oracle era sincronizzato con il database primario e prima che le applicazioni fossero riavviate nell'attività precedente.		DBA
Avvia le attività di AWS DMS in parallelo per replicare le modifiche in corso dal database di standby Oracle al database Aurora compatibile con PostgreSQL.		DBA

Operazione	Description	Competenze richieste
Ristabilisci la replica dal database primario Oracle al database di standby.		DBA
Monitora i log e arresta le applicazioni sul database Oracle quando il database di destinazione Aurora PostgreSQL compatibile è quasi sincrono con il database Oracle di origine.		DBA, proprietario dell'app
Interrompi le attività di AWS DMS quando la destinazione è completamente sincroniz zata con il database Oracle di origine.		DBA
Crea FKs e convalida i dati nel database di destinazione.		DBA
Crea funzioni, viste, trigger, sequenze e altri tipi di oggetti nel database di destinazione.		DBA
Applica le concessioni di ruolo nel database di destinazione.		DBA

Migrare l'applicazione

Operazione	Description	Competenze richieste
Usa AWS SCT per analizzare e convertire le istruzioni SQL		Proprietario dell'app

Operazione	Description	Competenze richieste
all'interno del codice dell'appl icazione.		
Crea nuovi server di applicazi oni su AWS.		Proprietario dell'app
Esegui la migrazione del codice dell'applicazione sui nuovi server.		Proprietario dell'app
Configura il server delle applicazioni per il database e i driver di destinazione.		Proprietario dell'app
Corregge qualsiasi codice specifico del motore di database di origine dell'appl icazione.		Proprietario dell'app
Ottimizza il codice dell'appl icazione per il database di destinazione.		Proprietario dell'app

Tagliare

Operazione	Description	Competenze richieste
Indirizza il nuovo server delle applicazioni verso il database di destinazione.		DBA, proprietario dell'app
Esegui controlli di integrità.		DBA, proprietario dell'app
Trasmetti in diretta.		DBA, proprietario dell'app

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse AWS temporanee.		DBA, amministratore di sistema
Rivedi e convalida i documenti del progetto.		DBA, proprietario dell'app
Raccogli le metriche relative al tempo di migrazione, alla percentuale di utilizzo manuale rispetto a quello degli strumenti, al risparmio sui costi e a dati simili.		DBA, proprietario dell'app
Chiudi il progetto e fornisci feedback.		DBA, proprietario dell'app

Risorse correlate

Riferimenti

- Compatibile con Oracle Database ad Aurora PostgreSQL: Migration Playbook
- Migrazione di un Amazon RDS per Oracle Database su Amazon Aurora MySQL
- Sito web AWS DMS
- Documentazione AWS DMS
- Sito web AWS SCT
- Documentazione AWS SCT
- Esegui la migrazione da Oracle ad Amazon Aurora

Tutorial

- Guida introduttiva ad AWS DMS
- Nozioni di base su Amazon RDS

- Step-by-StepProcedure dettagliate di AWS Database Migration Service

Esegui la migrazione da SAP ASE ad Amazon RDS per SQL Server utilizzando AWS DMS

Amit Kumar, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni per la migrazione di un database SAP Adaptive Server Enterprise (ASE) a un'istanza DB Amazon Relational Database Service (Amazon RDS) che esegue Microsoft SQL Server. Il database di origine può essere collocato in un data center locale o su un'istanza Amazon Elastic Compute Cloud EC2 (Amazon). Il modello utilizza AWS Database Migration Service (AWS DMS) per migrare i dati e (facoltativamente) strumenti di ingegneria del software assistita da computer (CASE) per convertire lo schema del database.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database SAP ASE in un data center locale o su un'istanza EC2
- Un database Amazon RDS for SQL Server di destinazione attivo e funzionante

Limitazioni

- Limite di dimensione del database: 64 TB

Versioni del prodotto

- Solo versione SAP ASE 15.7 o 16.x. Per le informazioni più recenti, consulta Using an SAP Database as a Source for AWS DMS.
- Per i database di destinazione Amazon RDS, AWS DMS supporta le versioni di Microsoft SQL Server su Amazon RDS per le edizioni Enterprise, Standard, Web ed Express. Per le informazioni più recenti sulle versioni supportate, consulta la documentazione di AWS DMS. Ti consigliamo di utilizzare la versione più recente di AWS DMS per il supporto più completo della versione e delle funzionalità.

Architecture

Stack tecnologico di origine

- Un database SAP ASE locale o su un'istanza Amazon EC2

Stack tecnologico Target

- Un'istanza DB di Amazon RDS per SQL Server

Architettura di origine e destinazione

Da un database SAP ASE su Amazon EC2 a un'istanza DB Amazon RDS for SQL Server:

Da un database SAP ASE locale a un'istanza DB Amazon RDS for SQL Server:

Tools (Strumenti)

- AWS Database Migration Service (AWS DMS) è un servizio Web che puoi utilizzare per migrare i dati dal tuo database locale, su un'istanza DB Amazon RDS o in un database su un' EC2 istanza, verso un database su un servizio AWS come Amazon RDS for SQL Server o un'istanza. EC2 Puoi anche migrare un database da un servizio AWS a un database locale. È possibile migrare i dati tra motori di database eterogenei o omogenei.
- Per le conversioni dello schema, puoi opzionalmente utilizzare erwin Data Modeler o SAP. PowerDesigner

Epiche

Pianifica la migrazione

Operazione	Description	Competenze richieste
Convalida le versioni del database di origine e di destinazione.		DBA

Operazione	Description	Competenze richieste
Identifica i requisiti di archiviazione (tipo e capacità di archiviazione).		DBA, SysAdmin
Scegli il tipo di istanza corretto in base alla capacità, alle funzionalità di archiviazione e alle funzionalità di rete.		DBA, SysAdmin
Identifica i requisiti di sicurezza dell'accesso alla rete per i database di origine e di destinazione.		DBA, SysAdmin
Identifica la strategia di migrazione delle applicazioni.		DBA SysAdmin, proprietario dell'app

Configura l'infrastruttura

Operazione	Description	Competenze richieste
Crea un cloud privato virtuale (VPC) e sottoreti.		SysAdmin
Crea gruppi di sicurezza e liste di controllo degli accessi alla rete (). ACLs		SysAdmin
Configura e avvia un'istanza database Amazon RDS.		SysAdmin

Migrazione dei dati - opzione 1

Operazione	Description	Competenze richieste
Esegui la migrazione manuale dello schema del database o utilizza uno strumento CASE come erwin Data Modeler o SAP. PowerDesigner		DBA

Migrazione dei dati - opzione 2

Operazione	Description	Competenze richieste
Migra i dati con AWS DMS.		DBA

Migrare l'applicazione

Operazione	Description	Competenze richieste
Segui la strategia di migrazion e delle applicazioni.		DBA SysAdmin, proprietario dell'app

Tagliare

Operazione	Description	Competenze richieste
Trasferisci i client applicativi alla nuova infrastruttura.		DBA SysAdmin, proprietario dell'app

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse AWS temporanee.		DBA, SysAdmin
Rivedi e convalida i documenti del progetto.		DBA, proprietario dell' SysAdminapp
Raccogli parametri come il tempo necessario per la migrazione, la percentuale di attività manuali rispetto a quelle automatizzate e il risparmio sui costi.		DBA, proprietario dell'app SysAdmin
Chiudi il progetto e fornisci feedback.		DBA SysAdmin, proprietario dell'app

Risorse correlate

Riferimenti

- Sito web AWS DMS
- Prezzi di Amazon RDS
- Utilizzo di un database SAP ASE come origine per AWS DMS
- Limitazioni per RDS Custom for SQL Server

Tutorial e video

- Guida introduttiva ad AWS DMS
- Nozioni di base su Amazon RDS
- AWS DMS (video)
- Amazon RDS (video)

Esegui la migrazione di un database Microsoft SQL Server locale su Amazon Redshift utilizzando AWS DMS

Marcelo Fernandes, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni per la migrazione di un database Microsoft SQL Server locale su Amazon Redshift utilizzando AWS Data Migration Service (AWS DMS).

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database Microsoft SQL Server di origine in un data center locale
- Prerequisiti completati per l'utilizzo di un database Amazon Redshift come destinazione per AWS DMS, come discusso nella documentazione di AWS DMS

Versioni del prodotto

- Edizioni SQL Server 2005-2019, Enterprise, Standard, Workgroup, Developer e Web. Per l'elenco più recente delle versioni supportate, consulta Using a Microsoft SQL Server Database as a Source for AWS DMS nella documentazione AWS.

Architecture

Stack tecnologico di origine

- Un database Microsoft SQL Server locale

Stack tecnologico Target

- Amazon Redshift

Architettura di migrazione dei dati

Tools (Strumenti)

- AWS DMS è un servizio di migrazione dei dati che supporta diversi tipi di database di origine e destinazione. Per informazioni sulle versioni e le edizioni del database Microsoft SQL Server supportate per l'uso con AWS DMS, consulta Using a Microsoft SQL Server Database as a Source for AWS DMS nella documentazione di AWS DMS. Se AWS DMS non supporta il tuo database di origine, devi selezionare un metodo alternativo per la migrazione dei dati.

Epiche

Pianifica la migrazione

Operazione	Description	Competenze richieste
Convalida la versione e il motore del database di origine e di destinazione.		DBA
Identifica i requisiti hardware per l'istanza del server di destinazione.		DBA, amministratore di sistema
Identifica i requisiti di archiviazione (tipo e capacità di archiviazione).		DBA, amministratore di sistema
Scegli il tipo di istanza corretto in base alla capacità, alle funzionalità di archiviazione e alle funzionalità di rete.		DBA, amministratore di sistema
Identifica i requisiti di sicurezza dell'accesso alla rete per i database di origine e di destinazione.		DBA, amministratore di sistema
Identifica la strategia di migrazione delle applicazioni.		DBA, proprietario dell'app, amministratore di sistema

Configura l'infrastruttura

Operazione	Description	Competenze richieste
Crea un cloud privato virtuale (Virtual Private Cloud (VPC))	Per ulteriori informazioni, consulta Lavorare con un'istanza DB in un VPC nella documentazione AWS.	Amministratore di sistema
Crea gruppi di sicurezza.		Amministratore di sistema
Configura e avvia un cluster Amazon Redshift.	Per ulteriori informazioni, consulta Creare un cluster Amazon Redshift di esempio nella documentazione di Amazon Redshift.	DBA, amministratore di sistema

Migrazione dei dati

Operazione	Description	Competenze richieste
Esegui la migrazione dei dati dal database Microsoft SQL Server utilizzando AWS DMS.		DBA

Migrare l'applicazione

Operazione	Description	Competenze richieste
Segui la strategia di migrazion e delle applicazioni.		DBA, proprietario dell'app, amministratore di sistema

Tagliare

Operazione	Description	Competenze richieste
Trasferisci i client applicativi alla nuova infrastruttura.		DBA, proprietario dell'app, amministratore di sistema

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse temporanee.		DBA, amministratore di sistema
Rivedi e convalida i documenti del progetto.		DBA, proprietario dell'app, amministratore di sistema
Raccogli parametri come il tempo necessario per la migrazione, la percentuale di attività manuali rispetto a quelle automatizzate e il risparmio sui costi.		DBA, proprietario dell'app, amministratore di sistema
Chiudi il progetto e fornisci feedback.		DBA, proprietario dell'app, amministratore di sistema

Risorse correlate

Riferimenti

- Documentazione AWS DMS
- Documentazione Amazon Redshift
- Prezzi di Amazon Redshift

Tutorial e video

- Guida introduttiva ad AWS DMS
- Nozioni di base su Amazon Redshift
- Utilizzo di un database Amazon Redshift come destinazione per AWS Database Migration Service
- AWS DMS (video)

Esegui la migrazione di un database Microsoft SQL Server locale su Amazon Redshift utilizzando gli agenti di estrazione dati AWS SCT

Neha Thakur, Amazon Web Services

Riepilogo

Questo modello descrive i passaggi per la migrazione di un database di origine Microsoft SQL Server locale a un database di destinazione Amazon Redshift utilizzando gli agenti di estrazione dei dati AWS Schema Conversion Tool (AWS SCT). Un agente è un programma esterno che è integrato con AWS SCT ma esegue la trasformazione dei dati altrove e interagisce con altri servizi AWS per tuo conto.

Prerequisiti e limitazioni

Prerequisiti

- Un database di origine Microsoft SQL Server utilizzato per il carico di lavoro del data warehouse in un data center locale
- Un account AWS attivo

Versioni del prodotto

- Microsoft SQL Server versione 2008 o successiva. Per l'elenco più recente delle versioni supportate, consulta la documentazione di AWS SCT.

Architecture

stack tecnologico Source

- Un database Microsoft SQL Server locale

stack tecnologico Target

- Amazon Redshift

Architettura di migrazione dei dati

Tools (Strumenti)

- AWS Schema Conversion Tool (AWS SCT) gestisce migrazioni di database eterogenei convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione. Quando i database di origine e di destinazione sono molto diversi, puoi utilizzare un agente AWS SCT per eseguire ulteriori trasformazioni dei dati. Per ulteriori informazioni, consulta Migrazione dei dati da un data warehouse locale ad Amazon Redshift nella documentazione AWS.

Best practice

- Le migliori pratiche per AWS SCT
- Best practice per Amazon Redshift

Epiche

Preparati per la migrazione

Operazione	Description	Competenze richieste
Convalida le versioni e i motori del database di origine e di destinazione.		DBA
Identifica i requisiti hardware per l'istanza del server di destinazione.		DBA, SysAdmin
Identifica i requisiti di archiviazione (tipo e capacità di archiviazione).		DBA, SysAdmin
Scegli il tipo di istanza corretto (capacità, funzionalità di archiviazione, funzionalità di rete).		DBA, SysAdmin

Operazione	Description	Competenze richieste
Identifica i requisiti di sicurezza dell'accesso alla rete per i database di origine e di destinazione.		DBA, SysAdmin
Scegli una strategia di migrazione delle applicazioni.		DBA SysAdmin, proprietario dell'app

Configurare l'infrastruttura

Operazione	Description	Competenze richieste
Crea un cloud privato virtuale (VPC) e sottoreti.		SysAdmin
Crea gruppi di sicurezza.		SysAdmin
Configura e avvia il cluster Amazon Redshift.		SysAdmin

Migrazione dei dati

Operazione	Description	Competenze richieste
Migra i dati utilizzando gli agenti di estrazione dati AWS SCT.		DBA

Migrazione delle applicazioni

Operazione	Description	Competenze richieste
Segui la strategia di migrazione delle applicazioni scelta.		DBA SysAdmin, proprietario dell'app

Passa al database di destinazione

Operazione	Description	Competenze richieste
Trasferisci i client applicativi alla nuova infrastruttura.		DBA SysAdmin, proprietario dell'app

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse AWS temporanee.		DBA, SysAdmin
Rivedi e convalida i documenti del progetto.		DBA, proprietario dell' SysAdminapp
Raccogli parametri come il tempo necessario per la migrazione, la percentuale di attività manuali rispetto a quelle automatizzate e il risparmio sui costi.		DBA, proprietario dell'app SysAdmin
Chiudi il progetto e fornisci qualsiasi feedback.		DBA SysAdmin, proprietario dell'app

Risorse correlate

Riferimenti

- Guida per l'utente di AWS SCT
- Utilizzo di agenti di estrazione dati
- Prezzi di Amazon Redshift

Tutorial e video

- Guida introduttiva allo Schema Conversion Tool di AWS
- Nozioni di base su Amazon Redshift

Migrazione delle applicazioni legacy da Oracle Pro*C a ECPG

Sai Parthasaradhi e Mahesh Balumuri, Amazon Web Services

Riepilogo

La maggior parte delle applicazioni legacy con codice SQL incorporato utilizza il precompilatore Oracle Pro*C per accedere al database. Quando esegui la migrazione di questi database Oracle ad Amazon Relational Database Service (Amazon RDS) per PostgreSQL o Amazon Aurora PostgreSQL Compatible Edition, devi convertire il codice dell'applicazione in un formato compatibile con il precompilatore di PostgreSQL, chiamato ECPG. Questo modello descrive come convertire il codice Oracle Pro*C nel suo equivalente in PostgreSQL ECPG.

Per ulteriori informazioni su Pro*C, consulta la documentazione Oracle. Per una breve introduzione a ECPG, vedere la sezione Informazioni aggiuntive.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database compatibile con Amazon RDS per PostgreSQL o Aurora PostgreSQL
- Un database Oracle in esecuzione in locale

Tools (Strumenti)

- I pacchetti PostgreSQL elencati nella sezione successiva.
- AWS CLI — L'AWS Command Line Interface (AWS CLI) è uno strumento open source per interagire con i servizi AWS tramite comandi nella shell della riga di comando. Con una configurazione minima, puoi eseguire comandi AWS CLI che implementano funzionalità equivalenti a quelle fornite dalla Console di gestione AWS basata su browser da un prompt dei comandi.

Epiche

Imposta l'ambiente di compilazione su CentOS o RHEL

Operazione	Description	Competenze richieste

Installa i pacchetti PostgreSQL.	Installa i pacchetti PostgreSQL richiesti utilizzando i seguenti comandi.
	`yum update -y`
	`yum install -y yum-utils`
	`rpm -ivh https://download.postgresql.org/pub/repos/yum/reporpms/EL-8-x86_64/pgdg-redhat-repo-latest.noarch.rpm`
	`dnf -qy module disable postgresql`
		Sviluppatore di app, ingegnere DevOps

Installa i file di intestazione e le librerie.	Installa il `postgresql12-devel` pacchetto, che contiene i file di intestazione e le librerie, utilizzando i seguenti comandi. Installa il pacchetto sia nell'ambiente di sviluppo che in quello di runtime per evitare errori nell'ambiente di runtime.
	`dnf -y install postgresql12-devel`
	`yum install ncompress zip ghostscript jq unzip wget git -y`
		Sviluppatore di app, DevOps ingegnere

Operazione	Description	Competenze richieste

	Solo per l'ambiente di sviluppo, esegui anche i seguenti comandi.
	`yum install zlib-devel make -y`
	`ln -s /usr/pgsql-12/bin/ecpg /usr/bin/`	

Configura la variabile del percorso di ambiente.	Imposta il percorso dell'ambiente per le librerie client PostgreSQL. `export PATH=$PATH:/usr/pgsql-12/bin`	Sviluppatore di app, ingegnere DevOps

Operazione	Description	Competenze richieste

Installa software aggiuntivo se necessario.	Se necessario, installare pgLoader in sostituzione di SQL*Loader in Oracle.
	`wget -O /etc/yum.repos.d/pgloader-ccl.repo https://dl.packager.io/srv/opf/pgloader-ccl/master/installer/el/7.repo`
	`yum install pgloader-ccl -y`
	`ln -s /opt/pgloader-ccl/bin/pgloader /usr/bin/`
	Se state chiamando delle applicazioni Java dai moduli Pro*C, installate Java.
	`yum install java -y`
	Installa ant per compilare il codice Java.
	`yum install ant -y`
		Sviluppatore di app, DevOps ingegnere

Operazione	Description	Competenze richieste

Installare l'interfaccia a riga di comando di AWS.	Installa l'AWS CLI per eseguire comandi per interagire con servizi AWS come AWS Secrets Manager e Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) dalle tue applicazioni.
	`cd /tmp/`
	`curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"`
	`unzip awscliv2.zip`
	`./aws/install -i /usr/local/aws-cli -b /usr/local/bin --update`
		Sviluppatore di app, ingegnere DevOps

Identifica i programmi da convertire.	Identifica le applicazioni che desideri convertire da Pro*C a ECPG.	Sviluppatore di app, proprietario dell'app

Converti il codice Pro*C in ECPG

Operazione	Description	Competenze richieste
Rimuovi le intestazioni indesiderate.	Rimuovi le `include` intestazioni che non sono richieste in PostgreSQL, ad esempio, e. `oci.h` `oratypes` `sqlda`	Proprietario dell'app, sviluppatore dell'app
Aggiorna le dichiarazioni delle variabili.	Aggiungi `EXEC SQL` istruzioni per tutte le dichiarazioni	Sviluppatore di app, proprietario dell'app

Operazione	Description	Competenze richieste
	di variabili utilizzate come variabili host. Rimuovi `EXEC SQL VAR` le dichiarazioni come le seguenti dall'applicazione. `EXEC SQL VAR query IS STRING(2048);`	

Operazione	Description	Competenze richieste

Aggiorna la funzionalità ROWNUM.	La `ROWNUM` funzione non è disponibile in PostgreSQL. Sostituiscila con la funzione `ROW_NUMBER` window nelle query SQL.
	
	Codice Pro*C:
	
	`SELECT SUBSTR(RT RIM(FILE_NAME,'.tx t'),12) INTO :gcpclFil eseq FROM (SELECT FILE_NAME FROM DEMO_FILES_TABLE WHERE FILE_NAME LIKE '%POC%' ORDER BY FILE_NAME DESC) FL2 WHERE ROWNUM <=1 ORDER BY ROWNUM;`
	
	Codice ECPG:
	
	`SELECT SUBSTR(RT RIM(FILE_NAME,'.tx t'),12) INTO :gcpclFil eseq FROM (SELECT FILE_NAME , ROW_NUMBE R() OVER (ORDER BY FILE_NAME DESC) AS ROWNUM FROM demo_sche ma.DEMO_FILES_TABLE WHERE FILE_NAME LIKE '%POC%' ORDER BY FILE_NAME DESC) FL2`
		Sviluppatore di app, proprieta rio dell'app

Operazione	Description	Competenze richieste
	`WHERE ROWNUM <=1 ORDER BY ROWNUM;`	

Aggiorna i parametri della funzione per utilizzare le variabili alias.	In PostgreSQL, i parametri delle funzioni non possono essere usati come variabili host. Sovrascrivili utilizzando una variabile alias.
	
	Codice Pro*C:
	
	`int processData(int referenceId){ EXEC SQL char col_val[100]; EXEC SQL select column_name INTO :col_val from table_name where col=:referenceId; }`
	
	Codice ECPG:
	
	`int processData(int referenceIdParam){ EXEC SQL int referenceId = referenceIdParam; EXEC SQL char col_val[100]; EXEC SQL select column_name INTO :col_val from table_name where col=:referenceId; }`
		Sviluppatore di app, proprietario dell'app

Operazione	Description	Competenze richieste
Aggiorna i tipi di struttura.	Definisci `struct` i tipi `EXEC SQL BEGIN` e `END` i blocchi specificando `typedef` se le variabili `struct` di tipo vengono utilizzate come variabili host. Se i `struct` tipi sono definiti nei file header (`.h`), includi i file con le istruzioni `EXEC SQL` include. Codice Pro*C: File di intestazione () `demo.h` `struct s_partition_ranges{ char sc_table_group[31]; char sc_table_name[31]; char sc_range_value[10];};struct s_partition_ranges_ind{ short ss_table_group; short ss_table_name; short ss_range_value;};` Codice ECPG: File di intestazione () `demo.h`	Sviluppatore di app, proprietario dell'app

Operazione	Description	Competenze richieste

```
EXEC SQL BEGIN DECLARE
 SECTION;
typedef struct
{
  char   sc_table_
group[31];
  char   sc_table_
name[31];
  char   sc_range_
value[10];
} s_partition_ranges;
typedef struct
{
  short    ss_table_
group;
  short    ss_table_
name;
  short    ss_range_
value;
} s_partition_ranges
_ind;
EXEC SQL END DECLARE
 SECTION;
```

File Pro*C () `demo.pc`

```
#include "demo.h"
struct s_partiti
on_ranges gc_partit
ion_data[MAX_PART_
TABLE] ;
struct s_partiti
on_ranges_ind
 gc_partition_data_
ind[MAX_PART_TABLE] ;
```

File ECPG () `demo.pc`

```
exec sql include
 "demo.h"
```

Operazione	Description	Competenze richieste
	`EXEC SQL BEGIN DECLARE SECTION; s_partition_ranges gc_partition_data[ MAX_PART_TABLE] ; s_partition_ranges_ind gc_partition_data_ ind[MAX_PART_TABLE] ; EXEC SQL END DECLARE SECTION;`	

Modifica la logica da recuperare dai cursori.	Per recuperare più righe dai cursori utilizzando variabili di matrice, modificate il codice da utilizzare. `FETCH FORWARD`
	
	Codice Pro*C:
	
	`EXEC SQL char aPoeFiles [MAX_FILES][FILENA ME_LENGTH]; EXEC SQL FETCH filename_ cursor into :aPoeFile s;`
	
	Codice ECPG:
	
	`EXEC SQL char aPoeFiles [MAX_FILES][FILENA ME_LENGTH]; EXEC SQL int fetchSize = MAX_FILES; EXEC SQL FETCH FORWARD :fetchSiz e filename_cursor into :aPoeFiles;`
		Sviluppatore di app, proprietario dell'app

Operazione	Description	Competenze richieste

Modifica le chiamate ai pacchetti che non hanno valori restituiti.	Le funzioni dei pacchetti Oracle che non hanno valori restituiti devono essere chiamate con una variabile indicatore. Se l'applicazione include più funzioni con lo stesso nome o se le funzioni di tipo sconosciuto generano errori di runtime, digita i valori nei tipi di dati.
	
	Codice Pro*C:
	
	`void ProcessData (char *data , int id) { EXEC SQL EXECUTE BEGIN pkg_demo.process_data (:data, :id); END; END-EXEC; }`
	
	Codice ECPG:
	
	`void ProcessData (char *dataParam, int idParam ) { EXEC SQL char *data = dataParam;`
		Sviluppatore di app, proprietario dell'app

Operazione	Description	Competenze richieste

	`EXEC SQL int id = idParam;`
	`EXEC SQL short rowInd;`
	`EXEC SQL short rowInd = 0;`
	`EXEC SQL SELECT pkg_demo.process_data (`
	`inp_data => :data::text,`
	`inp_id => :id )`
	`INTO :rowInd;`
	`}`	

Operazione	Description	Competenze richieste

Riscrivi le variabili SQL_CURSOR.	Riscrivi la variabile e la sua implementazione. SQL_CURSOR
	
	Codice Pro*C:
	
	`/* SQL Cursor */ SQL_CURSOR demo_cursor; EXEC SQL ALLOCATE :demo_cursor; EXEC SQL EXECUTE BEGIN pkg_demo. get_cursor( demo_cur= >:demo_cursor ); END; END-EXEC;`
	
	Codice ECPG:
	
	`EXEC SQL DECLARE demo_cursor CURSOR FOR SELECT * from pkg_demo.open_file name_rc( demo_cur= >refcursor ) ; EXEC SQL char open_file name_rcInd[100]; # As the below function returns cursor_name as # return we need to use char[] type as indicator.`
		Sviluppatore di app, proprietario dell'app

Operazione	Description	Competenze richieste
	`EXEC SQL SELECT pkg_demo.get_cursor ( demo_cur= >'demo_cursor' ) INTO :open_filename_rcInd;`	
Applica modelli di migrazione comuni.	• Modifica le query SQL in modo che siano compatibili con PostgreSQL. • Sposta i blocchi anonimi, quando non sono supportati in ECPG, nel database. • Rimuovi `dbms_application_info` la logica, che non è supportata da PostgreSQL. • Sposta `EXEC SQL COMMIT` le istruzioni dopo la chiusura del cursore. Se durante il ciclo si eseguono delle interrogazioni per recuperare i record dal cursore, il cursore viene chiuso e viene visualizzato l'errore «cursore non esiste». • Per informazioni sulla gestione delle eccezioni in ECPG e dei codici di errore, vedere Gestione degli errori nella documentazione di PostgreSQL.	Sviluppatore di app, proprietario dell'app

Operazione	Description	Competenze richieste
Abilita il debug, se necessario.	Per eseguire il programma ECPG in modalità debug, aggiungete il seguente comando all'interno del blocco funzionale principale. `ECPGdebug(1, stderr);`	Sviluppatore dell'app, proprietario dell'app

Compila programmi ECPG

Operazione	Description	Competenze richieste

Crea un file eseguibile per ECPG.	Se disponete di un file sorgente SQL C incorporato denominato `prog1.pgc`, potete creare un programma eseguibile utilizzando la seguente sequenza di comandi.
	`ecpg prog1.pgc`
	`cc -I/usr/local/pgsql/include -c prog1.c`
	`cc -o prog1 prog1.o -L/usr/local/pgsql/lib -lecpg`
		Sviluppatore di app, proprietario dell'app

Crea un make file per la compilazione.	Create un make file per compilare il programma ECPG, come mostrato nel seguente file di esempio. `CFLAGS ::= $(CFLAGS) -I/usr/pgsql-12/include -g -Wall`	Sviluppatore di app, proprietario dell'app

Operazione	Description	Competenze richieste

	`LDFLAGS ::= $(LDFLAGS) -L/usr/pgsql-12/lib -Wl,-rpath,/usr/pgsql-12/lib`
	`LDLIBS ::= $(LDLIBS) -lecpg`
	`PROGRAMS = test`
	`.PHONY: all clean`
	`%.c: %.pgc`
	`ecpg $<`
	`all: $(PROGRAMS)`
	`clean:`
	`rm -f $(PROGRAMS) $(PROGRAMS:%=%.c) $(PROGRAMS:%=%.o)`	

Eseguire il test dell'applicazione

Operazione	Description	Competenze richieste
Test del codice.	Verifica il codice dell'applicazione convertito per assicurarti che funzioni correttamente.	Sviluppatore di app, proprietario dell'app, tecnico di test

Risorse correlate

- ECPG - SQL integrato in C (documentazione PostgreSQL)
- Gestione degli errori (documentazione PostgreSQL)
- Perché utilizzare il precompilatore Oracle Pro*C/C++ (documentazione Oracle)

Informazioni aggiuntive

PostgreSQL ha un precompilatore SQL incorporato, ECPG, che è equivalente al precompilatore Oracle Pro*C. ECPG converte i programmi C che hanno istruzioni SQL incorporate in codice C

standard sostituendo le chiamate SQL con chiamate a funzioni speciali. I file di output possono quindi essere elaborati con qualsiasi catena di strumenti del compilatore C.

File di input e output

ECPG converte ogni file di input specificato nella riga di comando nel file di output C corrispondente. Se il nome di un file di input non ha un'estensione di file, viene utilizzato l'estensione.pgc. L'estensione del file viene sostituita da `.c` per costruire il nome del file di output. Tuttavia, è possibile sovrascrivere il nome del file di output predefinito utilizzando l'`-o`opzione.

Se utilizzate un trattino (`-`) come nome del file di input, ECPG legge il programma dallo standard input e scrive sullo standard output, a meno che non lo sovrascriviate utilizzando l'opzione. `-o`

File di intestazione

Quando il compilatore PostgreSQL compila i file di codice C preelaborati, cerca i file di intestazione ECPG nella directory PostgreSQL. `include` Pertanto, potrebbe essere necessario utilizzare l'`-I`opzione per indirizzare il compilatore alla directory corretta (ad esempio,). `-I/usr/local/pgsql/include`

Libraries (Librerie)

I programmi che utilizzano codice C con SQL incorporato devono essere collegati alla `libecpg` libreria. Ad esempio, è possibile utilizzare le opzioni `-L/usr/local/pgsql/lib -lecpg` del linker.

Le applicazioni ECPG convertite richiamano le funzioni della `libpq` libreria tramite la libreria SQL incorporata (`ecpglib`) e comunicano con il server PostgreSQL utilizzando il protocollo standard. frontend/backend

Migra le colonne virtuali generate da Oracle a PostgreSQL

Veeranjaneyulu Grandhi, Rajesh Madiwale e Ramesh Pathuri, Amazon Web Services

Riepilogo

Nella versione 11 e precedenti, PostgreSQL non fornisce una funzionalità direttamente equivalente a una colonna virtuale Oracle. La gestione delle colonne virtuali generate durante la migrazione da Oracle Database a PostgreSQL versione 11 o precedente è difficile per due motivi:

- Le colonne virtuali non sono visibili durante la migrazione.
- PostgreSQL non supporta `generate` l'espressione prima della versione 12.

Tuttavia, esistono soluzioni alternative per emulare funzionalità simili. Quando utilizzi AWS Database Migration Service (AWS DMS) per migrare i dati da Oracle Database a PostgreSQL versione 11 e precedenti, puoi utilizzare le funzioni di attivazione per popolare i valori in colonne virtuali generate. Questo modello fornisce esempi di codice Oracle Database e PostgreSQL che è possibile utilizzare per questo scopo. Su AWS, puoi utilizzare Amazon Relational Database Service (Amazon RDS) per PostgreSQL o Amazon Aurora PostgreSQL Compatible Edition per il tuo database PostgreSQL.

A partire dalla versione 12 di PostgreSQL, sono supportate le colonne generate. Le colonne generate possono essere calcolate istantaneamente in base ai valori di altre colonne oppure calcolate e archiviate. Le colonne generate da PostgreSQL sono simili alle colonne virtuali Oracle.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database Oracle di origine
- Database PostgreSQL di destinazione (su Amazon RDS per PostgreSQL o Aurora PostgreSQL compatibile)
- Esperienza nella codifica PL/PgSQL

Limitazioni

- Si applica solo alle versioni di PostgreSQL precedenti alla 12.
- Si applica alla versione 11g o successiva del database Oracle.

- Le colonne virtuali non sono supportate negli strumenti di migrazione dei dati.
- Si applica solo alle colonne definite nella stessa tabella.
- Se una colonna generata virtuale fa riferimento a una funzione deterministica definita dall'utente, non può essere utilizzata come colonna chiave di partizionamento.
- L'output dell'espressione deve essere un valore scalare. Non può restituire un tipo di dati fornito da Oracle, un tipo definito dall'utente o. `LOB LONG RAW`
- Gli indici definiti in base alle colonne virtuali sono equivalenti agli indici basati su funzioni in PostgreSQL.
- Le statistiche delle tabelle devono essere raccolte.

Tools (Strumenti)

- pgAdmin 4 è uno strumento di gestione open source per PostgreSQL. Questo strumento fornisce un'interfaccia grafica che semplifica la creazione, la manutenzione e l'uso degli oggetti del database.
- Oracle SQL Developer è un ambiente di sviluppo gratuito e integrato per lavorare con SQL nei database Oracle in implementazioni tradizionali e cloud.

Epiche

Crea tabelle di database di origine e di destinazione

Operazione	Description	Competenze richieste
Creare una tabella del database Oracle di origine.	In Oracle Database, crea una tabella con colonne virtuali generate utilizzando la seguente istruzione. `CREATE TABLE test.generated_column ( CODE NUMBER, STATUS VARCHAR2(12) DEFAULT 'PreOpen', FLAG CHAR(1) GENERATED ALWAYS AS (CASE UPPER(STATUS) WHEN`	DBA, sviluppatore di app

Operazione	Description	Competenze richieste

	<pre>'OPEN' THEN 'N' ELSE
	'Y' END) VIRTUAL
	VISIBLE
	);</pre>
	In questa tabella di origine, i dati nella `STATUS` colonna vengono migrati tramite AWS DMS al database di destinazione. La `FLAG` colonna, tuttavia, viene popolata utilizzando `generate by` funzionalità, quindi non è visibile ad AWS DMS durante la migrazione. Per implementare la funzionalità di`generated by`, è necessario utilizzare i trigger e le funzioni nel database di destinazione per compilare i valori nella `FLAG` colonna, come illustrato nella prossima epopea.	

Operazione	Description	Competenze richieste
Crea una tabella PostgreSQL di destinazione su AWS.	Crea una tabella PostgreSQL su AWS utilizzando la seguente istruzione. `CREATE TABLE test.generated_column ( code integer not null, status character varying(12) not null , flag character(1) );` In questa tabella, la `status` colonna è una colonna standard. La `flag` colonna sarà una colonna generata in base ai dati contenuti nella `status` colonna.	DBA, sviluppatore di app

Crea una funzione di attivazione per gestire la colonna virtuale in PostgreSQL

Operazione	Description	Competenze richieste
Crea un trigger PostgreSQL.	In PostgreSQL, crea un trigger. `CREATE TRIGGER tgr_gen_column AFTER INSERT OR UPDATE OF status ON test.generated_column FOR EACH ROW`	DBA, sviluppatore di app

Operazione	Description	Competenze richieste
	`EXECUTE FUNCTION test.tgf_gen_column();`	

Operazione	Description	Competenze richieste
Crea una funzione trigger PostgreSQL.	In PostgreSQL, crea una funzione per il trigger. Questa funzione popola una colonna virtuale che viene inserita o aggiornata dall'applicazione o da AWS DMS e convalida i dati. `CREATE OR REPLACE FUNCTION test.tgf_gen_column() RETURNS trigger AS $VIRTUAL_COL$ BEGIN IF (TG_OP = 'INSERT') THEN IF (NEW.flag IS NOT NULL) THEN RAISE EXCEPTION 'ERROR: cannot insert into column "flag"' USING DETAIL = 'Column "flag" is a generated column.'; END IF; END IF; IF (TG_OP = 'UPDATE') THEN IF (NEW.flag::VARCHAR != OLD.flag::varchar) THEN RAISE EXCEPTION 'ERROR: cannot update column "flag"' USING DETAIL = 'Column "flag" is a generated column.'; END IF; END IF; IF TG_OP IN ('INSERT','UPDATE') THEN`	DBA, sviluppatore di app

Operazione	Description	Competenze richieste
	`IF (old.flag is NULL) OR (coalesce(old.status,'') != coalesce(new.status,'')) THEN UPDATE test.generated_column SET flag = (CASE UPPER(status) WHEN 'OPEN' THEN 'N' ELSE 'Y' END) WHERE code = new.code; END IF; END IF; RETURN NEW; END $VIRTUAL_COL$ LANGUAGE plpgsql;`	

Testa la migrazione dei dati utilizzando AWS DMS

Operazione	Description	Competenze richieste
Creare un'istanza di replica.	Per creare un'istanza di replica, segui le istruzioni nella documentazione di AWS DMS. L'istanza di replica deve trovarsi nello stesso cloud privato virtuale (VPC) dei database di origine e di destinazione.	DBA, sviluppatore di app
Crea endpoint di origine e destinazione.	Per creare gli endpoint, segui le istruzioni nella documentazione di AWS DMS.	DBA, sviluppatore di app
Verifica le connessioni degli endpoint.	È possibile testare le connessioni degli endpoint	DBA, sviluppatore di app

Operazione	Description	Competenze richieste
	specificando il VPC e l'istanza di replica e scegliendo Esegui test.	
Crea e avvia un'attività a pieno carico.	Per istruzioni, consulta Creazione di un'attività e Impostazioni di caricamento completo nella documenta zione di AWS DMS.	DBA, sviluppatore di app
Convalida i dati per la colonna virtuale.	Confronta i dati nella colonna virtuale nei database di origine e di destinazione. È possibile convalidare i dati manualmen te o scrivere uno script per questo passaggio.	DBA, sviluppatore di app

Risorse correlate

- Guida introduttiva ad AWS Database Migration Service (documentazione AWS DMS)
- Utilizzo di un database Oracle come origine per AWS DMS (documentazione AWS DMS)
- Utilizzo di un database PostgreSQL come destinazione per AWS DMS (documentazione AWS DMS)
- Colonne generate in PostgreSQL (documentazione PostgreSQL)
- Funzioni di attivazione (documentazione PostgreSQL)
- Colonne virtuali in Oracle Database (documentazione Oracle)

Configura la funzionalità Oracle UTL_FILE su Aurora, compatibile con PostgreSQL

Rakesh Raghav e anuradha chintha, Amazon Web Services

Riepilogo

Durante il tuo percorso di migrazione da Oracle ad Amazon Aurora PostgreSQL Compatible Edition sul cloud Amazon Web Services (AWS), potresti incontrare diverse sfide. Ad esempio, la migrazione di codice che si basa sull'utilità Oracle è sempre una sfida. `UTL_FILE` In Oracle PL/SQL, il `UTL_FILE` pacchetto viene utilizzato per operazioni sui file, come lettura e scrittura, insieme al sistema operativo sottostante. L'`UTL_FILE`utilità funziona sia per i sistemi server che per quelli client.

Amazon Aurora PostgreSQL Compatible è un'offerta di database gestiti. Per questo motivo, non è possibile accedere ai file sul server del database. Questo modello illustra l'integrazione tra Amazon Simple Storage Service (Amazon S3) e la compatibilità con Amazon Aurora PostgreSQL per ottenere un sottoinsieme di funzionalità. `UTL_FILE` Grazie a questa integrazione, possiamo creare e consumare file senza utilizzare strumenti o servizi di estrazione, trasformazione e caricamento (ETL) di terze parti.

Facoltativamente, puoi configurare il CloudWatch monitoraggio di Amazon e le notifiche Amazon SNS.

Consigliamo di testare a fondo questa soluzione prima di implementarla in un ambiente di produzione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Esperienza in AWS Database Migration Service (AWS DMS)
- Esperienza nella codifica PL/pgSQL
- Un cluster compatibile con Amazon Aurora PostgreSQL
- Un bucket S3

Limitazioni

Questo modello non fornisce la funzionalità necessaria per sostituire l'utilità Oracle. `UTL_FILE` Tuttavia, i passaggi e il codice di esempio possono essere ulteriormente migliorati per raggiungere gli obiettivi di modernizzazione del database.

Versioni del prodotto

- Amazon Aurora versione 11.9 compatibile con PostgreSQL

Architecture

Stack tecnologico Target

- Compatibile con Amazon Aurora PostgreSQL
- Amazon CloudWatch
- Amazon Simple Notification Service (Amazon SNS)
- Simple Storage Service (Amazon S3)

Architettura Target

Il diagramma seguente mostra una rappresentazione di alto livello della soluzione.

1. I file vengono caricati dall'applicazione nel bucket S3.
2. L'`aws_s3`estensione accede ai dati, utilizzando PL/pgSQL, e carica i dati su Aurora PostgreSQL Compatible.

Tools (Strumenti)

- Compatibile con Amazon Aurora PostgreSQL — Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito, compatibile con PostgreSQL e conforme agli ACID. Combina la velocità e l'affidabilità dei database commerciali di fascia alta con l'economicità dei database open source.
- AWS CLI — L'AWS Command Line Interface (AWS CLI) è uno strumento unificato per gestire i servizi AWS. Con un solo strumento da scaricare e configurare, puoi controllare più servizi AWS dalla riga di comando e automatizzarli tramite script.
- Amazon CloudWatch: Amazon CloudWatch monitora le risorse e l'utilizzo di Amazon S3.

- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è uno storage per Internet. In questo modello, Amazon S3 fornisce un livello di storage per ricevere e archiviare file per il consumo e la trasmissione da e verso il cluster Aurora compatibile con PostgreSQL.
- aws_s3 — L'estensione `aws_s3` integra la compatibilità con Amazon S3 e Aurora PostgreSQL.
- Amazon SNS — Amazon Simple Notification Service (Amazon SNS) coordina e gestisce la consegna o l'invio di messaggi tra editori e clienti. In questo modello, Amazon SNS viene utilizzato per inviare notifiche.
- pgAdmin — pgAdmin è uno strumento di gestione open source per Postgres. pgAdmin 4 fornisce un'interfaccia grafica per la creazione, la manutenzione e l'utilizzo di oggetti di database.

Codice

Per ottenere la funzionalità richiesta, il pattern crea più funzioni con denominazione simile a. `UTL_FILE` La sezione Informazioni aggiuntive contiene il codice base per queste funzioni.

Nel codice, sostituiscilo `testaurorabucket` con il nome del bucket S3 di test. Sostituisci `us-east-1` con la regione AWS in cui si trova il bucket S3 di test.

Epiche

Integra la compatibilità con Amazon S3 e Aurora PostgreSQL

Operazione	Description	Competenze richieste
Configura le politiche IAM.	Crea policy AWS Identity and Access Management (IAM) che garantiscono l'accesso al bucket S3 e agli oggetti in esso contenuti. Per il codice, consulta la sezione Informazioni aggiuntive.	Amministratore AWS, DBA
Aggiungi i ruoli di accesso di Amazon S3 ad Aurora PostgreSQL.	Crea due ruoli IAM: un ruolo per l'accesso in lettura e un ruolo per l'accesso in scrittura ad Amazon S3. Collega i due ruoli al cluster compatibile con Aurora PostgreSQL:	Amministratore AWS, DBA

Operazione	Description	Competenze richieste
	• Un ruolo per la funzionalità S3Export • Un ruolo per la funzionalità S3Import Per ulteriori informazioni, consulta la documentazione compatibile con Aurora PostgreSQL sull'importazione e l'esportazione di dati su Amazon S3.	

Configura le estensioni in Aurora, compatibile con PostgreSQL

Operazione	Description	Competenze richieste
Crea l'estensione aws_commons.	L'`aws_commons` estensione è una dipendenza dell'estensione. `aws_s3`	DBA, Sviluppatore
Crea l'estensione aws_s3.	L'`aws_s3`estensione interagisce con Amazon S3.	DBA, Sviluppatore

Convalida l'integrazione compatibile con Amazon S3 e Aurora PostgreSQL

Operazione	Description	Competenze richieste
Prova a importare file da Amazon S3 in Aurora PostgreSQL.	Per testare l'importazione di file in Aurora PostgreSQL compatibile, crea un file CSV di esempio e caricalo nel bucket S3. Crea una definizione di tabella basata	DBA, Sviluppatore

Operazione	Description	Competenze richieste
	sul file CSV e carica il file nella tabella utilizzando la funzione. `aws_s3.table_import_from_s3`	
Prova a esportare file da Aurora PostgreSQL ad Amazon S3.	Per testare l'esportazione di file da Aurora PostgreSQL compatibile, crea una tabella di test, popolala con dati, quindi esporta i dati utilizzando la funzione. `aws_s3.query_export_to_s3`	DBA, Sviluppatore

Per imitare l'utilità UTL_FILE, create funzioni wrapper

Operazione	Description	Competenze richieste
Crea lo schema utl_file_utility.	Lo schema mantiene unite le funzioni del wrapper. Per creare lo schema, esegui il comando seguente. `CREATE SCHEMA utl_file_utility;`	DBA, Sviluppatore
Crea il tipo file_type.	Per creare il `file_type` tipo, utilizzate il codice seguente. `CREATE TYPE utl_file_utility.file_type AS ( p_path character varying(30), p_file_name character varying );`	DBA/Sviluppatore

Operazione	Description	Competenze richieste
Crea la funzione init.	La `init` funzione inizializza una variabile comune come o. `bucket` `region` Per il codice, consultate la sezione Informazioni aggiuntive.	DBA/Sviluppatore
Crea le funzioni wrapper.	Crea le funzioni `fopen` wrapper e. `put_line` `fclose` Per il codice, vedere la sezione Informazioni aggiuntive.	DBA, Sviluppatore

Prova le funzioni del wrapper

Operazione	Description	Competenze richieste
Prova le funzioni del wrapper in modalità scrittura.	Per testare le funzioni del wrapper in modalità scrittura, utilizzate il codice fornito nella sezione Informazioni aggiuntive.	DBA, Sviluppatore
Prova le funzioni del wrapper in modalità append.	Per testare le funzioni del wrapper in modalità di aggiunta, utilizzate il codice fornito nella sezione Informazioni aggiuntive.	DBA, Sviluppatore

Risorse correlate

- Integrazione con Amazon S3

- Amazon S3
- Aurora
- Amazon CloudWatch
- Amazon SNS

Informazioni aggiuntive

Configura le politiche IAM

Crea le seguenti politiche.

Nome policy	JSON

S3 IntRead	{
	"Version": "2012-10-17",
	"Statement": [
	{
	"Sid": "S3integrationtest
	",
	"Effect": "Allow",
	"Action": [
	"s3:GetObject",
	"s3:ListBucket"
	],
	"Resource": [
	"arn:aws:s3:::testaurorabuc
	ket/*",
	"arn:aws:s3:::testaurorabuc
	ket"
	]
	}
	]
	}

S3 IntWrite	{ "Version": "2012-10-17", "Statement": [{ "Sid": "S3integrationtest ",

```
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",

                "s3:ListBucket"
            ],
            "Resource": [
    "arn:aws:s3:::testaurorabucket/
*",               "arn:aws:s3:::test
aurorabucket"
            ]
        }
    ]
}
```

Crea la funzione init

Per inizializzare variabili comuni, come `bucket` `oregion`, create la `init` funzione utilizzando il codice seguente.

```
CREATE OR REPLACE FUNCTION utl_file_utility.init(
    )
    RETURNS void
    LANGUAGE 'plpgsql'

    COST 100
    VOLATILE
AS $BODY$
BEGIN
      perform set_config
      ( format( '%s.%s','UTL_FILE_UTILITY', 'region' )
      , 'us-east-1'::text
      , false );

      perform set_config
      ( format( '%s.%s','UTL_FILE_UTILITY', 's3bucket' )
      , 'testaurorabucket'::text
      , false );
END;
$BODY$;
```

Create le funzioni wrapper

Crea le funzioni `fopen`put_line, e `fclose` wrapper.

fopen

```
CREATE OR REPLACE FUNCTION utl_file_utility.fopen(
    p_file_name character varying,
    p_path character varying,
    p_mode character DEFAULT 'W'::bpchar,
    OUT p_file_type utl_file_utility.file_type)
    RETURNS utl_file_utility.file_type
    LANGUAGE 'plpgsql'

    COST 100
    VOLATILE
AS $BODY$
declare
    v_sql character varying;
    v_cnt_stat integer;
    v_cnt integer;
    v_tabname character varying;
    v_filewithpath character varying;
    v_region character varying;
    v_bucket character varying;

BEGIN
    /*initialize common variable */
    PERFORM utl_file_utility.init();
    v_region := current_setting( format( '%s.%s', 'UTL_FILE_UTILITY', 'region' ) );
    v_bucket :=  current_setting( format( '%s.%s', 'UTL_FILE_UTILITY', 's3bucket' ) );

    /* set tabname*/
    v_tabname := substring(p_file_name,1,case when strpos(p_file_name,'.') = 0 then
 length(p_file_name) else strpos(p_file_name,'.') - 1 end );
    v_filewithpath := case when NULLif(p_path,'') is null then p_file_name else
 concat_ws('/',p_path,p_file_name) end ;
    raise notice 'v_bucket %, v_filewithpath % , v_region %', v_bucket,v_filewithpath,
 v_region;

    /* APPEND MODE HANDLING; RETURN EXISTING FILE DETAILS IF PRESENT ELSE CREATE AN
 EMPTY FILE */
    IF p_mode = 'A' THEN
        v_sql := concat_ws('','create temp table if not exists ', v_tabname,' (col1
 text)');
        execute v_sql;
```

```
        begin
        PERFORM aws_s3.table_import_from_s3
            ( v_tabname,
            '',
            'DELIMITER AS ''#''',
            aws_commons.create_s3_uri
            (     v_bucket,
                v_filewithpath ,
                v_region)
            );
        exception
            when others then
             raise notice 'File load issue ,%',sqlerrm;
             raise;
        end;
        execute concat_ws('','select count(*) from ',v_tabname) into v_cnt;

        IF v_cnt > 0
        then
            p_file_type.p_path := p_path;
            p_file_type.p_file_name := p_file_name;
        else
            PERFORM aws_s3.query_export_to_s3('select ''''',
                            aws_commons.create_s3_uri(v_bucket, v_filewithpath,
 v_region)
                              );

            p_file_type.p_path := p_path;
            p_file_type.p_file_name := p_file_name;
        end if;
        v_sql := concat_ws('','drop table ', v_tabname);
        execute v_sql;
    ELSEIF p_mode = 'W' THEN
            PERFORM aws_s3.query_export_to_s3('select ''''',
                            aws_commons.create_s3_uri(v_bucket, v_filewithpath,
 v_region)
                              );
            p_file_type.p_path := p_path;
            p_file_type.p_file_name := p_file_name;
    END IF;

EXCEPTION
        when others then
```

```
            p_file_type.p_path := p_path;
            p_file_type.p_file_name := p_file_name;
            raise notice 'fopenerror,%',sqlerrm;
            raise;
END;
$BODY$;
```

put_line

```
CREATE OR REPLACE FUNCTION utl_file_utility.put_line(
    p_file_name character varying,
    p_path character varying,
    p_line text,
    p_flag character DEFAULT 'W'::bpchar)
    RETURNS boolean
    LANGUAGE 'plpgsql'

    COST 100
    VOLATILE
AS $BODY$
/*****************************************************************
* Write line, p_line in windows format to file, p_fp - with carriage return
* added before new line.
*****************************************************************/
declare
    v_sql varchar;
    v_ins_sql varchar;
    v_cnt INTEGER;
    v_filewithpath character varying;
    v_tabname  character varying;
    v_bucket character varying;
    v_region character varying;

BEGIN
 PERFORM utl_file_utility.init();

/* check if temp table already exist */

 v_tabname := substring(p_file_name,1,case when strpos(p_file_name,'.') = 0 then
 length(p_file_name) else strpos(p_file_name,'.') - 1 end );
```

```
 v_sql := concat_ws('','select count(1) FROM pg_catalog.pg_class c LEFT JOIN
 pg_catalog.pg_namespace n ON n.oid = c.relnamespace where n.nspname like ''pg_temp_
%'''
                         ,' AND pg_catalog.pg_table_is_visible(c.oid) AND
 Upper(relname) = Upper( '''
                         ,  v_tabname ,''' ) ');

 execute v_sql into v_cnt;

  IF v_cnt = 0 THEN
         v_sql := concat_ws('','create temp table ',v_tabname,' (col text)');
        execute v_sql;
        /* CHECK IF APPEND MODE */
        IF upper(p_flag) = 'A' THEN
            PERFORM utl_file_utility.init();
            v_region := current_setting( format( '%s.%s', 'UTL_FILE_UTILITY',
 'region' ) );
            v_bucket :=  current_setting( format( '%s.%s', 'UTL_FILE_UTILITY',
 's3bucket' ) );

            /* set tabname*/
            v_filewithpath := case when NULLif(p_path,'') is null then p_file_name else
 concat_ws('/',p_path,p_file_name) end ;

            begin
               PERFORM aws_s3.table_import_from_s3
                     ( v_tabname,
                          '',
                       'DELIMITER AS ''#''',
                        aws_commons.create_s3_uri
                           ( v_bucket,
                               v_filewithpath,
                               v_region    )
                    );
            exception
                when others then
                    raise notice  'Error Message : %',sqlerrm;
                    raise;
            end;
        END IF;
    END IF;
    /* INSERT INTO TEMP TABLE */
    v_ins_sql := concat_ws('','insert into ',v_tabname,' values(''',p_line,''')');
    execute v_ins_sql;
```

```
    RETURN TRUE;
    exception
            when others then
                raise notice  'Error Message : %',sqlerrm;
                raise;
END;
$BODY$;
```

chiudere

```
CREATE OR REPLACE FUNCTION utl_file_utility.fclose(
    p_file_name character varying,
    p_path character varying)
    RETURNS boolean
    LANGUAGE 'plpgsql'

    COST 100
    VOLATILE
AS $BODY$
DECLARE
    v_filewithpath character varying;
    v_bucket character varying;
    v_region character varying;
    v_tabname character varying;
    v_sql character varying;
BEGIN
      PERFORM utl_file_utility.init();

    v_region := current_setting( format( '%s.%s', 'UTL_FILE_UTILITY', 'region' ) );
    v_bucket :=  current_setting( format( '%s.%s', 'UTL_FILE_UTILITY', 's3bucket' ) );

    v_tabname := substring(p_file_name,1,case when strpos(p_file_name,'.') = 0 then
 length(p_file_name) else strpos(p_file_name,'.') - 1 end );
    v_filewithpath := case when NULLif(p_path,'') is null then p_file_name else
 concat_ws('/',p_path,p_file_name) end ;

    raise notice 'v_bucket %, v_filewithpath % , v_region %', v_bucket,v_filewithpath,
 v_region ;

    /* exporting to s3 */
    perform aws_s3.query_export_to_s3
        (concat_ws('','select * from ',v_tabname,'  order by ctid asc'),
            aws_commons.create_s3_uri(v_bucket, v_filewithpath, v_region)
```

```
        );
    v_sql := concat_ws('','drop table ', v_tabname);
    execute v_sql;
    RETURN TRUE;
EXCEPTION
       when others then
     raise notice 'error fclose %',sqlerrm;
     RAISE;
END;
$BODY$;
```

Metti alla prova le tue funzioni di configurazione e wrapper

Usa i seguenti blocchi di codice anonimi per testare la tua configurazione.

Prova la modalità di scrittura

Il codice seguente scrive un file denominato `s3inttest` nel bucket S3.

```
do $$
declare
l_file_name varchar := 's3inttest' ;
l_path varchar := 'integration_test' ;
l_mode char(1) := 'W';
l_fs utl_file_utility.file_type ;
l_status boolean;

begin
select * from
utl_file_utility.fopen( l_file_name, l_path , l_mode ) into l_fs ;
raise notice 'fopen : l_fs : %', l_fs;

select * from
utl_file_utility.put_line( l_file_name, l_path ,'this is test file:in s3bucket: for
 test purpose', l_mode ) into l_status ;
raise notice 'put_line : l_status %', l_status;

select * from utl_file_utility.fclose( l_file_name , l_path ) into l_status ;
raise notice 'fclose : l_status %', l_status;

end;
$$
```

Prova la modalità di aggiunta

Il codice seguente aggiunge righe al `s3inttest` file creato nel test precedente.

```
do $$
declare
l_file_name varchar := 's3inttest' ;
l_path varchar := 'integration_test' ;
l_mode char(1) := 'A';
l_fs utl_file_utility.file_type ;
l_status boolean;

begin
select * from
utl_file_utility.fopen( l_file_name, l_path , l_mode ) into l_fs ;
raise notice 'fopen : l_fs : %', l_fs;


select * from
utl_file_utility.put_line( l_file_name, l_path ,'this is test file:in s3bucket: for
 test purpose : append 1', l_mode ) into l_status ;
raise notice 'put_line : l_status %', l_status;

select * from
utl_file_utility.put_line( l_file_name, l_path ,'this is test file:in s3bucket : for
 test purpose : append 2', l_mode ) into l_status ;
raise notice 'put_line : l_status %', l_status;

select * from utl_file_utility.fclose( l_file_name , l_path ) into l_status ;
raise notice 'fclose : l_status %', l_status;

end;
$$
```

Notifiche Amazon SNS

Facoltativamente, puoi configurare il CloudWatch monitoraggio di Amazon e le notifiche Amazon SNS sul bucket S3. Per ulteriori informazioni, consulta Monitoraggio di Amazon S3 e Configurazione delle notifiche Amazon SNS.

Convalida gli oggetti del database dopo la migrazione da Oracle ad Amazon Aurora PostgreSQL

Venkatramana Chintha e Eduardo Valentim, Amazon Web Services

Riepilogo

Questo modello descrive un step-by-step approccio per convalidare gli oggetti dopo la migrazione di un database Oracle in Amazon Aurora PostgreSQL Compatible Edition.

Questo modello delinea gli scenari di utilizzo e le fasi per la convalida degli oggetti del database; per informazioni più dettagliate, consulta Convalida degli oggetti del database dopo la migrazione utilizzando AWS SCT e AWS DMS sul blog di AWS Database.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un database Oracle locale che è stato migrato a un database Aurora compatibile con PostgreSQL.
- Credenziali di accesso a cui è applicata la RDSData FullAccess policy di Amazon, per il database Aurora compatibile con PostgreSQL.
- Questo modello utilizza l'editor di query per i cluster DB Aurora Serverless, disponibile nella console Amazon Relational Database Service (Amazon RDS). Tuttavia, puoi utilizzare questo pattern con qualsiasi altro editor di query.

Limitazioni

- Gli oggetti Oracle SYNONYM non sono disponibili in PostgreSQL ma possono essere parzialmente convalidati tramite viste o query SET search_path.
- L'editor di query Amazon RDS è disponibile solo in alcune regioni AWS e per alcune versioni di MySQL e PostgreSQL.

Architecture

Tools (Strumenti)

Strumenti

- Amazon Aurora PostgreSQL Compatible Edition — Aurora PostgreSQL Compatible è un motore di database relazionale completamente gestito, compatibile con PostgreSQL e conforme ad ACID che combina la velocità e l'affidabilità dei database commerciali di fascia alta con la semplicità e l'economicità dei database open source.
- Amazon RDS — Amazon Relational Database Service (Amazon RDS) semplifica la configurazione, il funzionamento e la scalabilità di un database relazionale nel cloud AWS. Offre una capacità ridimensionabile a un costo conveniente per un database relazionale standard del settore e gestisce task comuni di amministrazione del database.
- Query Editor per Aurora Severless: l'editor di query ti aiuta a eseguire query SQL nella console Amazon RDS. È possibile eseguire qualsiasi istruzione SQL valida sul cluster DB Aurora Serverless, incluse le istruzioni di manipolazione e definizione dei dati.

Per convalidare gli oggetti, utilizza gli script completi nel file «Script di convalida degli oggetti» nella sezione «Allegati». Utilizzate la seguente tabella come riferimento.

Oggetto Oracle	Script da usare
Pacchetti	Query 1
Tabelle	Query 3
Visualizzazioni	Interrogazione 5
Sequenze	Interrogazione 7
Triggers	Interrogazione 9
Chiavi primarie	Interrogazione 11
Indici	Interrogazione 13
Vincoli check	Interrogazione 15
Chiavi esterne	Interrogazione 17

Oggetto PostgreSQL	Script da usare
Pacchetti	Query 2
Tabelle	Interrogazione 4
Visualizzazioni	Interrogazione 6
Sequenze	Interrogazione 8
Triggers	Interrogazione 10
Chiavi primarie	Interrogazione 12
Indici	Interrogazione 14
Vincoli check	Interrogazione 16
Chiavi esterne	Interrogazione 18

Epiche

Convalida gli oggetti nel database Oracle di origine

Operazione	Description	Competenze richieste
Esegui la query di convalida dei «pacchetti» nel database Oracle di origine.	Scarica e apri il file «Script di convalida degli oggetti» dalla sezione «Allegati». Connect al database Oracle di origine tramite il programma client. Esegui lo script di convalida «Query 1" dal file «Script di convalida degli oggetti». Importante: inserisci il tuo nome utente Oracle anziché «your_schema» nelle query.	Sviluppatore, DBA

Operazione	Description	Competenze richieste
	Assicurati di registrare i risultati delle query.	
Esegui la query di convalida «tables».	Esegui lo script «Query 3" dal file «Script di convalida degli oggetti». Assicurati di registrare i risultati della query.	Sviluppatore, DBA
Esegui la query di convalida «views».	Esegui lo script «Query 5" dal file «Script di convalida degli oggetti». Assicurati di registrare i risultati della query.	Sviluppatore, DBA
Esegui la convalida del conteggio delle «sequenze».	Esegui lo script «Query 7" dal file «Script di convalida degli oggetti». Assicurati di registrare i risultati della query.	Sviluppatore, DBA
Esegui la query di convalida «triggers».	Esegui lo script «Query 9" dal file «Script di convalida degli oggetti». Assicurati di registrare i risultati della query.	Sviluppatore, DBA
Esegui la query di convalida delle «chiavi primarie».	Esegui lo script «Query 11" dal file «Script di convalida degli oggetti». Assicurati di registrare i risultati della query.	Sviluppatore, DBA
Esegui la query di convalida degli «indici».	Esegui lo script di convalida «Query 13" dal file «Script di convalida degli oggetti». Assicurati di registrare i risultati della query.	Sviluppatore, DBA

Operazione	Description	Competenze richieste
Esegui la query di convalida «check constraints».	Esegui lo script «Query 15" dal file «Object validation scripts». Assicurati di registrar e i risultati della query.	Sviluppatore, DBA
Esegui la query di convalida delle «chiavi esterne».	Esegui lo script di convalida «Query 17" dal file «Script di convalida degli oggetti». Assicurati di registrare i risultati della query.	Sviluppatore, DBA

Convalida gli oggetti nel database Aurora di destinazione compatibile con PostgreSQL

Operazione	Description	Competenze richieste
Connect al database di destinazione compatibile con Aurora PostgreSQL utilizzan do l'editor di query.	Accedi alla Console di gestione AWS e apri la console Amazon RDS. Nell'angolo in alto a destra, scegli la regione AWS in cui hai creato il database Aurora compatibile con PostgreSQ L. Nel riquadro di navigazio ne, scegli «Database» e scegli il database di destinazi one compatibile con Aurora PostgreSQL. In «Azioni», scegli «Interrogazione». Importante: se non ti sei mai connesso al database prima, si apre la pagina «Connetti al database». È quindi necessari o inserire le informazioni del	Sviluppatore, DBA

Operazione	Description	Competenze richieste
	database, come nome utente e password.	
Esegui la query di convalida dei «pacchetti».	Esegui lo script «Query 2" dal file «Script di convalida degli oggetti» nella sezione «Allegati». Assicurati di registrare i risultati della query.	Sviluppatore, DBA
Esegui la query di convalida «tables».	Tornate all'editor di query per il database Aurora compatibile con PostgreSQL ed eseguite lo script «Query 4" dal file «Object validation scripts». Assicurati di registrare i risultati della query.	Sviluppatore, DBA
Esegui la query di convalida «views».	Tornate all'editor di query per il database Aurora compatibile con PostgreSQL ed eseguite lo script «Query 6" dal file «Object validation scripts». Assicurati di registrare i risultati della query.	Sviluppatore, DBA
Esegui la convalida del conteggio delle «sequenze».	Tornate all'editor di query per il database Aurora compatibile con PostgreSQL ed eseguite lo script «Query 8" dal file «Object validation scripts». Assicurati di registrare i risultati della query.	Sviluppatore, DBA

Operazione	Description	Competenze richieste
Esegui la query di convalida «triggers».	Tornate all'editor di query per il database Aurora compatibile con PostgreSQL ed eseguite lo script «Query 10" dal file «Object validation scripts». Assicurati di registrare i risultati della query.	Sviluppatore, DBA
Esegui la query di convalida delle «chiavi primarie».	Tornate all'editor di query per il database Aurora compatibile con PostgreSQL ed eseguite lo script «Query 12" dal file «Object validation scripts». Assicurati di registrare i risultati della query.	Sviluppatore, DBA
Esegui la query di convalida degli «indici».	Tornate all'editor di query per il database Aurora compatibile con PostgreSQL ed eseguite lo script «Query 14" dal file «Object validation scripts». Assicurati di registrare i risultati della query.	Sviluppatore, DBA
Esegui la query di convalida «check constraints».	Esegui lo script «Query 16" dal file «Object validation scripts». Assicurati di registrar e i risultati della query.	Sviluppatore, DBA
Esegui la query di convalida delle «chiavi esterne».	Esegui lo script di convalida «Query 18" dal file «Object validation scripts». Assicurat i di registrare i risultati della query.	Sviluppatore, DBA

Confronta i record di convalida del database di origine e di destinazione

Operazione	Description	Competenze richieste
Confronta e convalida entrambi i risultati delle query.	Confronta i risultati delle query dei database compatibili con Oracle e Aurora PostgreSQL per convalidare tutti gli oggetti. Se tutti corrispondono, allora tutti gli oggetti sono stati convalidati correttamente.	Sviluppatore, DBA

Risorse correlate

- Convalida degli oggetti del database dopo una migrazione utilizzando AWS SCT e AWS DMS
- Caratteristiche di Amazon Aurora: edizione compatibile con PostgreSQL

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Riospitare

Argomenti

- Accelera la scoperta e la migrazione dei carichi di lavoro Microsoft su AWS
- Creare un processo di approvazione per le richieste del firewall durante una migrazione di rehosting a AWS
- Acquisisci e migra istanze EC2 Windows in un account AWS Managed Services
- Esegui la migrazione di un database Couchbase Server su Amazon EC2
- Esegui la migrazione di Db2 for LUW ad Amazon EC2 utilizzando la spedizione di log per ridurre i tempi di interruzione
- Esegui la migrazione di Db2 per LUW ad Amazon EC2 con disaster recovery ad alta disponibilità
- Esegui la migrazione di applicazioni ospitate da IIS su Amazon utilizzando appcmd.exe EC2
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon EC2 utilizzando Application Migration Service
- Esegui la migrazione di un carico di lavoro F5 BIG-IP su F5 BIG-IP VE su Cloud AWS
- Esegui la migrazione di un'applicazione web Go locale su AWS Elastic Beanstalk utilizzando il metodo binario
- Esegui la migrazione di un server SFTP locale a Using AWS AWS Transfer for SFTP
- Migra una macchina virtuale locale su Amazon EC2 utilizzando AWS Application Migration Service
- Esegui la migrazione di piccoli set di dati da locale ad Amazon S3 utilizzando AWS SFTP
- Esegui la migrazione di un database Oracle locale a Oracle su Amazon EC2
- Esegui la migrazione di un database Oracle locale su Amazon EC2 utilizzando Oracle Data Pump
- Migra i sistemi RHEL BYOL verso istanze con licenza AWS inclusa utilizzando AWS MGN
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon EC2
- Rehosting dei carichi di lavoro locali nel cloud AWS: lista di controllo per la migrazione
- Configura un'infrastruttura Multi-AZ per SQL Server Always On FCI utilizzando Amazon FSx
- Usa le query BMC Discovery per estrarre i dati di migrazione per la pianificazione della migrazione

Accelera la scoperta e la migrazione dei carichi di lavoro Microsoft su AWS

Ali Alzand, Amazon Web Services

Riepilogo

Questo modello mostra come utilizzare il PowerShell modulo Migration Validator Toolkit per scoprire e migrare i carichi di lavoro Microsoft su AWS. Il modulo funziona eseguendo più controlli e convalide per le attività comuni associate a qualsiasi carico di lavoro Microsoft. Ad esempio, il modulo verifica le istanze che potrebbero avere più dischi collegati o le istanze che utilizzano molti indirizzi IP. Per un elenco completo dei controlli che il modulo può eseguire, consulta la sezione Controlli nella pagina del modulo. GitHub

Il PowerShell modulo Migration Validator Toolkit può aiutare l'organizzazione a ridurre il tempo e l'impegno necessari per scoprire quali applicazioni e servizi sono in esecuzione sui carichi di lavoro Microsoft. Il modulo può anche aiutarti a identificare le configurazioni dei tuoi carichi di lavoro per scoprire se le tue configurazioni sono supportate su AWS. Il modulo fornisce anche consigli sui passaggi successivi e sulle operazioni di mitigazione, in modo da evitare configurazioni errate prima, durante o dopo la migrazione.

Prerequisiti e limitazioni

Prerequisiti

- Account amministratore locale
- PowerShell 4.0

Limitazioni

- Funziona solo su Microsoft Windows Server 2012 R2 o versioni successive

Tools (Strumenti)

Strumenti

- PowerShell 4.0

Deposito di codici

Il PowerShell modulo Migration Validator Toolkit per questo modello è disponibile nell'archivio - microsoft-workloads. GitHub migration-validator-toolkit-for

Epiche

Esegui il PowerShell modulo Migration Validator Toolkit su un singolo target

Operazione	Description	Competenze richieste

Scarica, estrai, importa e richiama il modulo.	Scegliete uno dei seguenti metodi per scaricare e distribuire il modulo:
	• Esegui lo script PowerShell
	• Scarica ed estrai il file.zip
	• Clona il repository GitHub
	
	Esegui lo script PowerShell
	
	In PowerShell, esegui il seguente codice di esempio:
	
	`#MigrationValidatorToolkit`
	`$uri = 'https://github.com/aws-samples/migration-validator-toolkit-for-microsoft-workloads/archive/refs/heads/main.zip'`
	`$destination = (Get-Location).Path`
	`if ((Test-Path -Path "$destination\MigrationValidatorToolkit.zip" -PathType Leaf) -or (Test-Path -Path "$destination\MigrationValidatorToolkit")) {`
		Amministratore di sistema

Operazione	Description	Competenze richieste
	write-host "File $destination\MigrationValidatorToolkit.zip or folder $destination\MigrationValidatorToolkit found, exiting" }else { Write-host "Enable TLS 1.2 for this PowerShell session only." [Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12 $webClient = New-Object System.Net.WebClient Write-host "Downloading Migration ValidatorToolkit.zip" $webClient.DownloadFile($uri, "$destination\MigrationValidatorToolkit.zip") Write-host "MigrationValidatorToolkit.zip download successfully" Add-Type -Assembly "system.io.compression.filesystem" [System.IO.Compression.ZipFile]::ExtractToDirectory("$destination\MigrationValidatorToolkit.zip","$destination\MigrationValidatorToolkit")	

Operazione	Description	Competenze richieste

	<pre>Write-host
	"Extracting Migration
	ValidatorToolkit.zip
	complete successfully"
	Import-Module
	"$destination\Migr
	ationValidatorToolkit
	\migration-validator-
	toolkit-for-microsoft
	-workloads-main\Mi
	grationValidatorTo
	olkit.psm1"; Invoke-
	MigrationValidatorTo
	olkit
	}</pre>
	Il codice scarica il modulo da un file.zip. Quindi, il codice estrae, importa e richiama il modulo.
	Scarica ed estrai il file.zip
	1. Scarica il file.zip (download).
	2. Decomprimere il file .zip.
	3. Segui i passaggi descritti nella storia Invoke the module manual di questa guida.
	Clona il repository GitHub
	1. Per clonare l'archivio GitHub migration-validator-toolkit-for-microsoft-workloads, esegui il seguente	

Operazione	Description	Competenze richieste
	comando Git in una finestra di terminale: `git clone https://github.com/aws-samples/migration-validator-toolkit-for-microsoft-workloads.git` 2. Segui i passaggi descritti nella storia Invoke the module manual di questa guida.	

Operazione	Description	Competenze richieste

Richiama il modulo manualmente.	1. Vai alla directory in cui è memorizzato il modulo scaricato.
	2. Per generare l'output desiderato, esegui uno dei seguenti comandi come amministratore in PowerShell:
	
	Formato tabella-formato:
	
	`Import-Module .\MigrationValidatorToolkit.psm1;Invoke-MigrationValidatorToolkit`
	
	Formato dell'elenco dei formati:
	
	`Import-Module .\MigrationValidatorToolkit.psm1;Invoke-MigrationValidatorToolkit -List`
	
	Formato di uscita: GridView
	
	`Import-Module .\MigrationValidatorToolkit.psm1;Invoke-MigrationValidatorToolkit -GridView`
	
	ConvertTo-Formato csv:
		Amministratore di sistema

Operazione	Description	Competenze richieste
	`Import-Module .\MigrationValidatorToolkit.psm1;Invoke-MigrationValidatorToolkit -csv`	

Esegui il modulo Migration Validator Toolkit PowerShell su più destinazioni

Operazione	Description	Competenze richieste
Scarica il file.zip o clona il GitHub repository.	Scegli una delle seguenti opzioni: • Scarica il file zip. (scarica). • Per clonare l'archivio GitHub migration-validator-toolkit-for-microsoft-workloads, esegui il seguente comando Git in una finestra di terminale: `git clone https://github.com/aws-samples/migration-validator-toolkit-for-microsoft-workloads.git`	Amministratore di sistema
Aggiorna l'elenco server.csv.	Se hai scaricato il file.zip, procedi nel seguente modo: 1. Decomprimere il file .zip.	Amministratore di sistema

Operazione	Description	Competenze richieste
	2. Vai alla directory `MigrationValidator Toolkit\Inputs\ .` 3. Aggiorna `serverlis t.csv` con il nome host dei computer di destinazi one.	

Operazione	Description	Competenze richieste
Invoca il modulo.	È possibile utilizzare qualsiasi computer all'interno del dominio che utilizza un utente di dominio con accesso amministratore ai computer di destinazione. 1. Scarica il codice sorgente come file.zip ed estrai il file. 2. Come amministratore di PowerShell, esegui il seguente comando: `Import-Module .\MigrationValidatorToolkit.psm1;Invoke-DomainComputers` Il file.csv di output viene salvato `Migration ValidatorToolkit \Outputs\folder` con il nome del prefisso. `DomainComputers_MigrationAutomations_YYYY-MM-DDTHH-MM-SS`	Amministratore di sistema

risoluzione dei problemi

Problema	Soluzione
`MigrationValidatorToolkit` scrive informazioni su esecuzioni, comandi ed errori nei file di registro sull'host in esecuzione.	È possibile visualizzare i file di registro manualmente nella seguente posizione: 1. Vai alla directory `MigrationValidatorToolkit\logs\` . 2. Individua il file di registro. Il formato del nome del file di registro è: `ComputerName_MigrationValidatorToolkit_YYYY-MM-SSTHH-MM-SS.log`

Risorse correlate

- Opzioni, strumenti e best practice per la migrazione dei carichi di lavoro Microsoft su AWS (AWS Prescriptive Guidance)
- Modelli di migrazione Microsoft (AWS Prescriptive Guidance)
- Servizi di migrazione cloud gratuiti su AWS (documentazione AWS)
- Azioni predefinite dopo il lancio (documentazione di marketing delle applicazioni)

Informazioni aggiuntive

Domande frequenti

Dove posso eseguire il modulo Migration Validator Toolkit? PowerShell

È possibile eseguire il modulo su Microsoft Windows Server 2012 R2 o versioni successive.

Quando posso eseguire questo modulo?

Ti consigliamo di eseguire il modulo durante la fase di valutazione del percorso di migrazione.

Il modulo modifica i miei server esistenti?

No. Tutte le azioni in questo modulo sono di sola lettura.

Quanto tempo occorre per eseguire il modulo?

L'esecuzione del modulo richiede in genere da 1 a 5 minuti, ma dipende dall'allocazione delle risorse del server.

Di quali autorizzazioni ha bisogno il modulo per funzionare?

È necessario eseguire il modulo da un account amministratore locale.

Posso eseguire il modulo su server fisici?

Sì, purché il sistema operativo sia Microsoft Windows Server 2012 R2 o versione successiva.

Come posso eseguire il modulo su larga scala per più server?

Per eseguire il modulo su più computer aggiunti a un dominio su larga scala, segui i passaggi indicati nel modulo Esegui il PowerShell modulo Run the Migration Validator Toolkit su più obiettivi di questa guida. Per i computer non aggiunti al dominio, utilizzate una chiamata remota o eseguite il modulo localmente seguendo i passaggi descritti nel modulo Esegui il modulo Esegui il modulo Migration Validator Toolkit su un singolo obiettivo di questa guida. PowerShell

Creare un processo di approvazione per le richieste del firewall durante una migrazione di rehosting a AWS

Srikanth Rangavajhala, Amazon Web Services

Riepilogo

Se desideri utilizzare AWS Application Migration Serviceo attivare Cloud Migration Factory AWS per una migrazione di rehosting verso Cloud AWS, uno dei prerequisiti è mantenere aperte le porte TCP 443 e 1500. In genere, l'apertura di queste porte firewall richiede l'approvazione del team di sicurezza delle informazioni (). InfoSec

Questo modello delinea il processo per ottenere l'approvazione di una richiesta di firewall da parte di un InfoSec team durante una migrazione di rehost verso. Cloud AWSÈ possibile utilizzare questo processo per evitare che la richiesta relativa al firewall venga respinta dal InfoSec team, operazione che può diventare costosa e dispendiosa in termini di tempo. Il processo di richiesta del firewall prevede due fasi di revisione e approvazione tra consulenti di AWS migrazione e responsabili che collaborano con i team addetti all'applicazione InfoSec e all'apertura delle porte del firewall.

Questo modello presuppone che stiate pianificando una migrazione di rehosting con AWS consulenti o specialisti della migrazione della vostra organizzazione. È possibile utilizzare questo modello se l'organizzazione non dispone di un processo di approvazione del firewall o di un modulo di approvazione generale per la richiesta del firewall. Per ulteriori informazioni su questo argomento, consulta la sezione Limitazioni di questo modello. Per ulteriori informazioni sui requisiti di rete per Application Migration Service, consulta Requisiti di rete nella documentazione di Application Migration Service.

Prerequisiti e limitazioni

Prerequisiti

- Una migrazione riospitata pianificata con AWS consulenti o specialisti della migrazione della vostra organizzazione
- Le informazioni necessarie sulla porta e sull'IP per migrare lo stack
- Diagrammi delle architetture di stato esistenti e future
- Informazioni sul firewall relative all'infrastruttura, alle porte e al flusso di traffico locali e di destinazione zone-to-zone
- Una lista di controllo per la revisione delle richieste di firewall (in allegato)
- Un documento di richiesta del firewall, configurato in base ai requisiti dell'organizzazione

- Un elenco di contatti per i revisori e gli approvatori del firewall, che include i seguenti ruoli:
  - Richiedente della richiesta relativa al firewall: AWS specialista o consulente in materia di migrazione. Il mittente della richiesta di firewall può anche essere uno specialista della migrazione della vostra organizzazione.
  - Firewall Request Reviewer: in genere, si tratta del punto di contatto unico (SPOC) di. AWS
  - Firewall Request Approver: un membro del team InfoSec .

Limitazioni

- Questo modello descrive un processo generico di approvazione delle richieste di firewall. I requisiti possono variare a seconda delle singole organizzazioni.
- Assicurati di tenere traccia delle modifiche al documento di richiesta del firewall.

La tabella seguente mostra i casi d'uso di questo pattern.

La tua organizzazione dispone di un processo di approvazione del firewall esistente?	La tua organizzazione dispone già di un modulo di richiesta per il firewall?	Azione suggerita
Sì	Sì	Collabora con AWS consulenti o specialisti della migrazione per implementare il processo della tua organizzazione.
No	Sì	Utilizza il processo di approvazione del firewall di questo pattern. Rivolgiti a un AWS consulente o a uno specialista della migrazione della tua organizzazione per inviare il modulo di approvazione generale della richiesta del firewall.
No	No	Utilizzate il processo di approvazione del firewall di

	questo pattern. Rivolgiti a un AWS consulente o a uno specialista della migrazione della tua organizzazione per inviare il modulo di approvazione generale della richiesta del firewall.

Architecture

Il diagramma seguente mostra i passaggi del processo di approvazione della richiesta del firewall.

Tools (Strumenti)

È possibile utilizzare strumenti di scansione come Palo Alto Networks o SolarWindsper analizzare e convalidare firewall e indirizzi IP.

Epiche

Analizza la richiesta del firewall

Operazione	Description	Competenze richieste
Analizza le porte e gli indirizzi IP.	Il mittente della richiesta di firewall completa un'analisi iniziale per comprendere le porte e gli indirizzi IP del firewall richiesti. Al termine, richiedono che il InfoSec team apra le porte richieste e mappi gli indirizzi IP.	Ingegnere del cloud AWS, specialista della migrazione

Convalida la richiesta del firewall

Operazione	Description	Competenze richieste
Convalida le informazioni sul firewall.	L' Cloud AWS ingegnere pianifica un incontro con il tuo InfoSec team. Durante questa riunione, l'ingegnere esamina e convalida le informazioni sulla richiesta del firewall. In genere, chi invia la richiesta del firewall è la stessa persona del richiedente del firewall. Questa fase di convalida può diventare iterativa in base al feedback fornito dall'appr ovatore se qualcosa viene osservato o consigliato.	Ingegnere del cloud AWS, specialista della migrazione
Aggiorna il documento di richiesta del firewall.	Dopo che il InfoSec team ha condiviso il feedback, il documento di richiesta del firewall viene modificato, salvato e ricaricato. Questo documento viene aggiornato dopo ogni iterazione. Si consiglia di archiviare questo documento in una cartella di archiviazione controllata dalla versione. Ciò significa che tutte le modifiche vengono tracciate e applicate correttamente.	Ingegnere del cloud AWS, specialista della migrazione

Invia la richiesta relativa al firewall

Operazione	Description	Competenze richieste
Invia la richiesta del firewall.	Dopo che l'approvatore della richiesta firewall ha approvato la richiesta di approvazi one generale del firewall, il Cloud AWS tecnico invia la richiesta al firewall. La richiesta specifica le porte che devono essere aperte e gli indirizzi IP necessari per mappare e aggiornare il. Account AWS È possibile fornire suggerime nti o fornire feedback dopo l'invio della richiesta del firewall. Ti consigliamo di automatizzare questo processo di feedback e di inviare eventuali modifiche tramite un meccanismo di flusso di lavoro definito.	Ingegnere del cloud AWS, specialista della migrazione

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Acquisisci e migra istanze EC2 Windows in un account AWS Managed Services

Anil Kunapareddy e Venkatramana Chintha, Amazon Web Services

Riepilogo

Questo modello spiega il step-by-step processo di migrazione e acquisizione di istanze Windows di Amazon Elastic Compute Cloud (EC2Amazon) in un account Amazon Web Services (AWS) Managed Services (AMS). AMS può aiutarti a gestire l'istanza in modo più efficiente e sicuro. AMS offre flessibilità operativa, migliora la sicurezza e la conformità e aiuta a ottimizzare la capacità e ridurre i costi.

Questo modello inizia con un'istanza di EC2 Windows che hai migrato a una sottorete di staging nel tuo account AMS. Sono disponibili diversi servizi e strumenti di migrazione per eseguire questa attività, come AWS Application Migration Service.

Per apportare una modifica al tuo ambiente gestito da AMS, devi creare e inviare una richiesta di modifica (RFC) per una particolare operazione o azione. Utilizzando un RFC AMS workload ingest (WIGS), si inserisce l'istanza nell'account AMS e si crea un'Amazon Machine Image (AMI) personalizzata. Quindi crei l' EC2 istanza gestita da AMS inviando un'altra RFC per creare uno stack. EC2 Per ulteriori informazioni, consulta AMS Workload Ingest nella documentazione AMS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo e gestito da AMS
- Una landing zone esistente
- Autorizzazioni per apportare modifiche nel VPC gestito da AMS
- Un'istanza Amazon EC2 Windows in una sottorete di staging nel tuo account AMS
- Completamento dei prerequisiti generali per la migrazione dei carichi di lavoro utilizzando AMS WIGS
- Completamento dei prerequisiti di Windows per la migrazione dei carichi di lavoro utilizzando AMS WIGS

Limitazioni

- Questo modello si riferisce alle EC2 istanze che utilizzano Windows Server. Questo modello non si applica alle istanze che eseguono altri sistemi operativi, come Linux.

Architecture

Stack di tecnologia di origine

Istanza Amazon EC2 Windows in una sottorete di staging nel tuo account AMS

Stack tecnologico Target

Istanza Amazon EC2 Windows gestita da AWS Managed Services (AMS)

Architettura Target

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi usare Amazon EC2 per lanciare tutti o pochi server virtuali di cui hai bisogno, con scalabilità orizzontale o orizzontale.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Managed Services (AMS) ti aiuta a operare in modo più efficiente e sicuro fornendo una gestione continua della tua infrastruttura AWS, tra cui monitoraggio, gestione degli incidenti, guida alla sicurezza, supporto di patch e backup per i carichi di lavoro AWS.

Altri servizi

- PowerShellè un programma di gestione dell'automazione e della configurazione di Microsoft che funziona su Windows, Linux e macOS.

Epiche

Configura le impostazioni sull'istanza

Operazione	Description	Competenze richieste
Modifica le impostazioni del client DNS.	1. Nell' EC2 istanza di origine, apri il prompt dei	Ingegnere della migrazione

Operazione	Description	Competenze richieste
	comandi come amministratore **`gpedit.msc`** , digita e premi Invio. 2. Nel Local Group Policy Editor, accedi a Configurazione computer, Modelli amministrativi, Rete, Client DNS. 3. Per il suffisso DNS primario, scegli Non configurato. 4. Per la devoluzione del suffisso DNS primario, scegli Non configurato.	
Modifica le impostazioni di Windows Update.	1. Nel Local Group Policy Editor, accedi a Configurazione computer, Modelli amministrativi, Componenti di Windows, Windows Update. 2. Per Specificare la posizione del servizio di aggiornamento Microsoft Intranet, scegli Non configurato. 3. Per Configura aggiornamenti automatici, scegli Non configurato. 4. Per la frequenza di rilevamento degli aggiornamenti automatici, scegli Non configurato. 5. Chiudi il Local Group Policy Editor.	Ingegnere della migrazione

Operazione	Description	Competenze richieste
Attiva il firewall.	1. Nell' EC2 istanza di origine, apri il prompt dei comandi come amministratore`services.msc` , digita e premi Invio. 2. In Windows Services, abilita Firewall. 3. Chiudi i servizi Windows.	Tecnico di migrazione

Prepara l'istanza per AMS WIGS

Operazione	Description	Competenze richieste
Pulisci e prepara l'istanza.	1. Utilizzando un host bastion e credenziali locali, crea una connessione RDP (Remote Desktop Protocol) all' EC2 istanza nella sottorete di staging. 2. Rimuovi tutti i software legacy, i software antivirus e le soluzioni di backup che non sono richiesti in AMS.	Ingegnere della migrazione
Ripara il file sppnp.dll.	1. Passa a `C:\Windows\System32\sppnp.dll` . 2. Rinomina `sppnp.dll` in. `sppnp_old.dll` 3. Utilizzando PowerShell le credenziali di amministratore, immettete i seguenti comandi:	Tecnico di migrazione

Operazione	Description	Competenze richieste
	``` dism /online /cleanup-image /restorehealth sfc /scannnow ``` 4. Riavviare l'istanza EC2 di Windows.	
Esegui lo script di convalida pre-WIG.	1. Scarica il file zip di convalida pre-ingestione di Windows WIGS (`windows-prewings-valication.zip`) da Migrating workloads: Windows pre-ingestion validation nella documentazione AMS. 2. Esegui lo script di convalida Windows Pre-WIG e verifica i risultati. 3. Se la convalida fallisce, risolvi il problema ed esegui nuovamente lo script di convalida fino al completamento della convalida.	Ingegnere della migrazione

Operazione	Description	Competenze richieste
Crea l'AMI failsafe.	Una volta superata la convalida pre-WIG, crea un'AMI di pre-ingestione come segue: 1. Scegli Deployment, Advanced stack components, AMI, Create. 2. Durante la creazione, aggiungi un tag `Key=Name,Value=APPLICATION-ID_IngestReady` . 3. Attendi la creazione dell'AMI prima di procedere. Per ulteriori informazioni, consulta AMI | Create nella documentazione AMS.	Ingegnere della migrazione

Acquisisci e convalida l'istanza

Operazione	Description	Competenze richieste
Invia la RFC per creare lo stack di acquisizione del carico di lavoro.	Invia una richiesta di modifica (RFC) per avviare AMS WIGS. Per istruzioni, consulta Workload Ingest Stack: Creating nella documentazione AMS. Questo avvia l'acquisizione del carico di lavoro e installa tutto il software richiesto da AMS, inclusi gli strumenti di backup,	Ingegnere della migrazione

Operazione	Description	Competenze richieste
	il software di EC2 gestione Amazon e il software antivirus.	
Convalida la migrazione riuscita.	Una volta completata l'acquisizione del carico di lavoro, puoi visualizzare l'istanza gestita da AMS e l'AMI inserita da AMS. 1. Accedi all'istanza gestita da AMS con le credenziali del dominio. 2. Convalida l'aggiunta al dominio come segue: a. In Windows Explorer, fai clic con il pulsante destro del mouse su Questo PC, quindi scegli Proprietà. b. Nella sezione Specifiche del dispositivo, verifica che il dominio sia visualizzato nel Nome completo del dispositivo. 3. Convalida le unità disco di origine e di destinazione.	Ingegnere della migrazione

Avvia l'istanza nell'account AMS di destinazione

Operazione	Description	Competenze richieste
Invia la RFC per creare uno EC2 stack.	1. Utilizzando l'AMI inserita da AMS dell'istanza Windows, prepara una RFC per EC2 uno stack in base	Ingegnere della migrazione

Operazione	Description	Competenze richieste
	alle istruzioni in EC2 Create stack instance nella documentazione AMS. Nello EC2 stack RFC, fornisci tutti i parametri, inclusi il nome del server, i tag, il VPC di destinazione, la sottorete di destinazione, il tipo di istanza, i gruppi di sicurezza di destinazione, l'AMI di importazione e il ruolo. 2. Invia la RFC per lo EC2 stack, quindi attendi che l'istanza venga creata correttamente.	

Risorse correlate

Prontuario AWS

- Automatizza le attività di pre-inserimento del carico di lavoro per AWS Managed Services su Windows
- Crea automaticamente un RFC in AMS usando Python

Documentazione AMS

- Inserimento del carico di lavoro AMS
- In che modo la migrazione cambia la tua risorsa
- Migrazione dei carichi di lavoro: processo standard

Risorse di marketing

- AWS Managed Services

- AWS Managed Services FAQs
- Risorse AWS Managed Services
- Caratteristiche di AWS Managed Services

Esegui la migrazione di un database Couchbase Server su Amazon EC2

Subhani Shaik, Amazon Web Services

Riepilogo

Questo modello descrive come migrare Couchbase Server da un ambiente locale ad Amazon Elastic Compute Cloud (Amazon) on. EC2 AWS

Couchbase Server è un database NoSQL (documento JSON) distribuito che fornisce funzionalità di database relazionali. La migrazione di un database Couchbase Server a AWS può fornire maggiore scalabilità, migliori prestazioni, efficienza dei costi, maggiore sicurezza, gestione semplificata e portata globale, a vantaggio delle applicazioni che richiedono alta disponibilità e accesso ai dati a bassa latenza. È inoltre possibile accedere a funzionalità avanzate tramite servizi gestiti. AWS

Couchbase Server on AWS offre le seguenti funzionalità chiave:

- Architettura incentrata sulla memoria
- Alta disponibilità, disaster recovery e bilanciamento del carico
- Implementazione multimaster e multiregione per prestazioni ottimali

Per ulteriori informazioni sui vantaggi principali, consulta la sezione Informazioni aggiuntive e il sito Web di Couchbase.

Prerequisiti e limitazioni

Prerequisiti

- Un sistema attivo Account AWS con un cloud privato virtuale (VPC), due zone di disponibilità, sottoreti private e un gruppo di sicurezza. Per istruzioni, consulta Creare un VPC nella documentazione di Amazon Virtual Private Cloud (Amazon VPC).
- Connettività abilitata tra ambienti di origine e destinazione. Per informazioni sulle porte TCX utilizzate da Couchbase Server, consulta la documentazione di Couchbase.

Architecture

Il diagramma seguente mostra l'architettura di alto livello per la migrazione di Couchbase Server a. AWS

Dal cluster Couchbase locale, i dati si spostano attraverso un gateway per il cliente utilizzando. AWS Direct Connect I dati passano attraverso un router e un Direct Connect percorso e raggiungono il VPC tramite un gateway AWS Virtual Private Network (Site-to-Site VPN). Il VPC contiene un' EC2 istanza che esegue Couchbase Server. L' AWS infrastruttura include anche AWS Identity and Access Management (IAM) per il controllo degli accessi, AWS Key Management Service (AWS KMS) per la crittografia dei dati, Amazon Elastic Block Store (Amazon EBS) per lo storage a blocchi e Amazon Simple Storage Service (Amazon S3) per lo storage dei dati.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- AWS Direct Connectcollega la rete interna a una Direct Connect posizione tramite un cavo Ethernet standard in fibra ottica. Con questa connessione, è possibile creare interfacce virtuali direttamente al pubblico Servizi AWS ignorando i provider di servizi Internet nel percorso di rete.

Best practice

- Installazione e configurazione di Couchbase su diverse piattaforme operative
- Le migliori pratiche per la distribuzione di Couchbase Server su AWS
- Creazione di un cluster Couchbase
- Best practice in materia di prestazioni per le applicazioni Couchbase
- Le migliori pratiche di sicurezza per Couchbase Server
- Le migliori pratiche di archiviazione per i database di Couchbase Server

Epiche

Implementa un' EC2 istanza Amazon per Couchbase Server

Operazione	Description	Competenze richieste
Apri la EC2 console Amazon.	Accedi Console di gestione AWSe apri la EC2 console Amazon.	DevOps ingegnere, amministratore di Couchbase

Operazione	Description	Competenze richieste
Implementa un' EC2 istanza Amazon.	Avvia un' EC2 istanza che corrisponda alle configura zioni del server Couchbase locale. Per ulteriori informazi oni su come distribuire un' EC2 istanza, consulta Launch an Amazon EC2 istance nella EC2 documentazione di Amazon.	DevOps ingegnere, amministr atore di Couchbase

Installa e configura Couchbase Server su Amazon EC2

Operazione	Description	Competenze richieste
Installa un cluster Couchbase.	Consulta le linee guida per la distribuzione di Couchbase Server prima di installar e Couchbase Server su Amazon. EC2 Per installare Couchbase Server, consulta la documenta zione di Couchbase Server	Amministratore di Couchbase
Configura il cluster.	Per configurare il cluster, consulta le opzioni di configurazione del cluster nella documentazione di Couchbase.	Amministratore di Couchbase

Aggiungi un nuovo nodo e ribilancia il cluster Couchbase

Operazione	Description	Competenze richieste
Aggiungi un nodo per l' EC2 istanza.	Aggiungi l' EC2 istanza appena distribuita su cui è installato Couchbase al cluster locale esistente. Per istruzioni, consulta Aggiungi un nodo e ribilanciamento nella documentazione del server Couchbase.	Amministratore di Couchbase
Ribilanciare il cluster.	Il processo di ribilanciamento rende il nodo appena aggiunto con l' EC2 istanza un membro attivo del cluster Couchbase . Per istruzioni, consulta Aggiungi un nodo e ribilanciamento nella documentazione del server Couchbase	Amministratore di Couchbase

Riconfigurazione delle connessioni

Operazione	Description	Competenze richieste
Rimuovi i nodi locali e ribilanciali.	È ora possibile rimuovere i nodi locali dal cluster. Dopo aver rimosso i nodi, segui il processo di ribilanciamento per ridistribuire dati, indici, elaborazione degli eventi e elaborazione delle query tra i nodi disponibili nel cluster. Per istruzioni, consulta Rimuovere un nodo e ribilanciare nella	Amministratore di Couchbase

Operazione	Description	Competenze richieste
	documentazione del server Couchbase.	
Aggiorna i parametri di connessione.	Aggiorna i parametri di connessione dell'applicazione per utilizzare il nuovo indirizzo EC2 IP Amazon, in modo che l'applicazione possa connetter si al nuovo nodo.	Sviluppatore di applicazioni Couchbase

Risorse correlate

- Servizi Couchbase Server
- Distribuisci Couchbase Server utilizzando Marketplace AWS
- Connect a Couchbase Server
- Gestisci i bucket
- Replica tra data center (XDCR)
- Contratto di licenza di Couchbase Inc.

Informazioni aggiuntive

Vantaggi principali

La migrazione del database Couchbase a offre i seguenti vantaggi: AWS

Scalabilità. È possibile aumentare o ridurre il cluster Couchbase in base alla richiesta senza dover gestire l'hardware fisico, in modo da poter gestire facilmente i volumi di dati e l'utilizzo delle applicazioni fluttuanti. AWS fornisce:

- opzioni di ridimensionamento verticale e orizzontale
- Funzionalità di implementazione globali
- Bilanciamento del carico su tutto Regioni AWS
- Soluzioni di scalabilità del database
- Ottimizzazione della distribuzione dei contenuti

Ottimizzazione delle prestazioni. AWS fornisce un'infrastruttura di rete ad alte prestazioni e tipi di istanze ottimizzati per garantire un accesso rapido ai dati e una bassa latenza per il database Couchbase.

- Opzioni di elaborazione ad alte prestazioni (HPC)
- Distribuzione globale di contenuti tramite Amazon CloudFront
- Molteplici opzioni di archiviazione
- Servizi di database avanzati, tra cui Amazon Relational Database Service (Amazon RDS) e Amazon DynamoDB
- Connessioni a bassa latenza con Direct Connect

Ottimizzazione dei costi. Seleziona il tipo e la configurazione di istanza appropriati per bilanciare prestazioni e costi in base al carico di lavoro. Paghi solo per le risorse che utilizzi. Ciò può potenzialmente ridurre i costi operativi eliminando la necessità di gestire l'hardware locale e sfruttando Cloud AWS le economie di scala.

- Le istanze riservate possono aiutarti a pianificare in anticipo e ridurre sostanzialmente i costi quando utilizzi Couchbase su. AWS
- La scalabilità automatica impedisce l'eccessivo approvvigionamento e aiuta a ottimizzare l'utilizzo e l'efficienza in termini di costi.

Sicurezza migliorata. Approfitta delle solide funzionalità di sicurezza disponibili AWS, come la crittografia dei dati, i controlli di accesso e i gruppi di sicurezza per proteggere i dati sensibili archiviati in Couchbase. Vantaggi aggiuntivi:

- Il modello di responsabilitàAWS condivisa distingue chiaramente tra sicurezza del cloud (AWS responsabilità) e sicurezza nel cloud (responsabilità del cliente).
- AWS la conformità supporta i principali standard di sicurezza.
- AWS offre opzioni di crittografia avanzate.
- AWS Identity and Access Management (IAM) ti aiuta a gestire l'accesso sicuro alle tue risorse.

Gestione semplificata. AWS fornisce servizi gestiti per Couchbase, quindi puoi concentrarti sullo sviluppo di applicazioni anziché sulla gestione dell'infrastruttura sottostante.

Copertura globale. Puoi distribuire il tuo cluster Couchbase su più cluster Regioni AWS per ottenere una bassa latenza per gli utenti di tutto il mondo. Puoi distribuire i tuoi database interamente nel cloud o in un ambiente ibrido. Puoi proteggere i tuoi dati con una sicurezza integrata di livello aziendale e una sincronizzazione bidirezionale rapida ed efficiente dei dati dall'edge al cloud. Allo stesso tempo, puoi semplificare lo sviluppo con un modello di programmazione coerente per la creazione di app web e mobili.

Continuità aziendale:

- Backup e ripristino dei dati. In caso di problemi, è possibile utilizzare AWS Backupper garantire la resilienza dei dati e un facile ripristino. Per le opzioni di disaster recovery, consulta la documentazione di AWS Well-Architected Framework.
- Distribuzione multiregionale di Couchbase: per distribuire un database Couchbase in un AWS ambiente multiregionale, puoi abbonarti a Couchbase Server in Marketplace AWS, utilizzare AWS CloudFormationmodelli per creare cluster Couchbase separati in ciascuna regione e quindi configurare la replica interregionale per sincronizzare i dati tra le regioni. Questa configurazione garantisce un'elevata disponibilità e ridondanza geografica tra più regioni. Per ulteriori informazioni, consulta Distribuire Couchbase Server Using Marketplace AWS nella documentazione di Couchbase.

Agilità dell'infrastruttura:

- Approvvigionamento e disattivazione rapidi delle risorse
- Copertura globale dell'infrastruttura
- Scalabilità automatica in base alla domanda
- Infrastructure as Code (IaC) per implementazioni coerenti
- Diversi tipi di istanze ottimizzate per carichi di lavoro diversi

Abilitazione dell'innovazione:

- Accesso alle tecnologie più recenti, tra cui AI/ML, IoT e analisi
- Servizi gestiti, che riducono il sovraccarico operativo
- Pratiche moderne di sviluppo di applicazioni
- Opzioni di elaborazione serverless

Eccellenza operativa:

- Monitoraggio e registrazione centralizzati
- Gestione automatizzata delle risorse
- Funzionalità di manutenzione predittiva
- Visibilità migliorata sull'utilizzo delle risorse
- Processi di implementazione semplificati

Opportunità di modernizzazione:

- Architettura di microservizi
- DevOpsimplementazione delle pratiche
- sviluppo di applicazioni native per il cloud
- Modernizzazione delle applicazioni legacy

Vantaggi competitivi:

- Tempi di commercializzazione più rapidi
- Esperienza del cliente migliorata
- Processo decisionale basato sui dati
- Business intelligence migliorata

Esegui la migrazione di Db2 for LUW ad Amazon EC2 utilizzando la spedizione di log per ridurre i tempi di interruzione

Feng Cai, Ambarish Satarkar e Saurabh Sharma, Amazon Web Services

Riepilogo

Quando i clienti migrano i loro carichi di lavoro IBM Db2 for LUW (Linux, UNIX e Windows) su Amazon Web Services (AWS), utilizzare Amazon Elastic Compute Cloud (Amazon) con il modello Bring Your Own License (BYOL) è il modo più veloce. EC2 Tuttavia, la migrazione di grandi quantità di dati da Db2 locale ad AWS può essere una sfida, soprattutto quando la finestra di interruzione è breve. Molti clienti cercano di impostare la finestra di interruzione su meno di 30 minuti, il che lascia poco tempo per il database stesso.

Questo modello illustra come eseguire una migrazione Db2 con una breve finestra di interruzione utilizzando la spedizione dei log delle transazioni. Questo approccio si applica a Db2 su una piattaforma Linux little-endian.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un'istanza Db2 in esecuzione su EC2 un'istanza che corrisponde ai layout del file system locale
- Un bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) accessibile all'istanza EC2
- Una policy e un ruolo di AWS Identity and Access Management (IAM) per effettuare chiamate programmatiche ad Amazon S3
- Fuso orario e orologi di sistema sincronizzati su Amazon EC2 e sul server locale
- La rete locale connessa ad AWS tramite AWS Site-to-SiteVPN o AWS Direct Connect

Limitazioni

- L'istanza locale Db2 e Amazon EC2 devono appartenere alla stessa famiglia di piattaforme.
- Il carico di lavoro locale di Db2 deve essere registrato. Per bloccare qualsiasi transazione non registrata, impostala nella configurazione del database. `blocknonlogged=yes`

Versioni del prodotto

- Db2 per LUW versione 11.5.9 e successive

Architecture

Stack tecnologico di origine

- Db2 su Linux x86_64

Stack tecnologico Target

- Amazon EBS
- Amazon EC2
- AWS Identity and Access Management (IAM)
- Simple Storage Service (Amazon S3)
- AWS Site-to-Site VPN o Direct Connect

Architettura Target

Il diagramma seguente mostra un'istanza Db2 in esecuzione in locale con una connessione di rete privata virtuale (VPN) a Db2 su Amazon. EC2 Le linee tratteggiate rappresentano il tunnel VPN tra il tuo data center e il cloud AWS.

Tools (Strumenti)

Servizi AWS

- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.
- AWS Direct Connect collega la rete interna a una posizione Direct Connect tramite un cavo Ethernet standard in fibra ottica. Con questa connessione, puoi creare interfacce virtuali direttamente ai servizi AWS pubblici bypassando i provider di servizi Internet nel tuo percorso di rete.
- Amazon Elastic Block Store (Amazon EBS) fornisce volumi di storage a livello di blocco da utilizzare con istanze Amazon Elastic Compute Cloud (Amazon). EC2
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.

- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Site-to-Site VPN ti aiuta a trasferire il traffico tra le istanze che avvii su AWS e la tua rete remota.

Altri strumenti

- db2cli è il comando CLI interattivo di Db2.

Best practice

- Sul database di destinazione, utilizza gli endpoint gateway per Amazon S3 per accedere all'immagine di backup del database e ai file di log in Amazon S3.
- Sul database di origine, usa AWS PrivateLink per Amazon S3 per inviare l'immagine di backup del database e i file di log ad Amazon S3.

Epiche

Impostazione delle variabili di ambiente

Operazione	Description	Competenze richieste
Imposta le variabili di ambiente.	Questo modello utilizza i seguenti nomi: • nome dell'istanza: `db2inst1` • Nome del database: `SAMPLE` È possibile modificarli per adattarli al proprio ambiente.	DBA

Configura il server Db2 locale

Operazione	Description	Competenze richieste

Configurare AWS CLI.	Per scaricare e installare la versione più recente dell'interfaccia a riga di comando di AWS, esegui i seguenti comandi:
	`$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"`
	`unzip awscliv2.zip`
	`sudo ./aws/install`
		Amministratore Linux

Configura una destinazione locale per i log di archivio Db2.	Per mantenere il database di destinazione su Amazon EC2 sincronizzato con il database di origine locale, è necessario recuperare i log delle transazioni più recenti dall'origine. In questa configurazione, `/db2logs` viene impostato da `LOGARCHMETH2` on the source come area di staging. I log archiviati in questa directory verranno sincronizzati in Amazon S3 e vi si accederà da Db2 su Amazon. EC2 Il pattern utilizza `LOGARCHMETH2` because `LOGARCHMETH1` potrebbe essere stato configurato per utilizzare uno strumento di	DBA

Operazione	Description	Competenze richieste
	un fornitore di terze parti a cui il comando AWS CLI non può accedere. Per recuperare i log, esegui il seguente comando: `db2 connect to sample db2 update db cfg for SAMPLE using LOGARCHMETH2 disk:/db2logs`	
Esegui un backup del database online.	Esegui un backup del database online e salvalo nel file system di backup locale: `db2 backup db sample online to /backup`	DBA

Configura il bucket S3 e la policy IAM

Operazione	Description	Competenze richieste
Crea un bucket S3.	Crea un bucket S3 per il server locale a cui inviare le immagini Db2 di backup e i file di log su AWS. Il bucket sarà accessibile anche da Amazon EC2: `aws s3api create-bucket --bucket logshipmig-db2 --region us-east-1`	Amministratore di sistema AWS

Operazione	Description	Competenze richieste
Creare una policy IAM	Il `db2bucket.json` file contiene la policy IAM per accedere al bucket Amazon S3: `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [ "kms:GenerateDataKey", "kms:Decrypt", "s3:PutObject", "s3:GetObject", "s3:AbortMultipartUpload", "s3:ListBucket", "s3:DeleteObject", "s3:GetObjectVersion", "s3:ListMultipartUploadParts" ], "Resource": [`	Amministratore AWS, amministratore di sistema AWS

Operazione	Description	Competenze richieste

	<pre>"arn:aws:s3:::logshipmig-db2/*",
	
	"arn:aws:s3:::logshipmig-db2"
	]
	}
	]
	}</pre>
	Per creare la policy, usa il seguente comando AWS CLI:
	<pre>aws iam create-policy \
	--policy-name db2s3policy \
	--policy-document file://db2bucket.json</pre>
	L'output JSON mostra l'Amazon Resource Name (ARN) per la policy, `aws_account_id` dove rappresenta l'ID del tuo account:
	<pre>"Arn": "arn:aws:iam::aws_account_id:policy/db2s3policy"</pre>	

Operazione	Description	Competenze richieste
Collega la policy IAM al ruolo IAM utilizzato dall' EC2 istanza.	Nella maggior parte degli ambienti AWS, un' EC2 istanza in esecuzione ha un ruolo IAM impostato dall'amministratore di sistema. Se il ruolo IAM non è impostato, crea il ruolo e scegli Modifica il ruolo IAM sulla EC2 console per associare il ruolo all' EC2 istanza che ospita il database Db2. Collega la policy IAM al ruolo IAM con la policy ARN: `aws iam attach-role-policy \ --policy-arn "arn:aws:iam::aws_account_id:policy/db2s3policy" \ --role-name db2s3role` Dopo aver allegato la policy, qualsiasi EC2 istanza associata al ruolo IAM può accedere al bucket S3.	Amministratore AWS, amministratore di sistema AWS

Inviare l'immagine di backup del database di origine e i file di log ad Amazon S3

Operazione	Description	Competenze richieste
Configura l'AWS CLI sul server Db2 locale.	Configura la CLI AWS con `Access Key ID` e `Secret Access Key` generata nel passaggio precedente:	Amministratore AWS, amministratore di sistema AWS

Operazione	Description	Competenze richieste

	`$ aws configure`
	`AWS Access Key ID`
	` [None]: ************`
	`AWS Secret Access Key`
	` [None]: *********`
	`**********`
	`Default region name`
	` [None]: us-east-1`
	`Default output format`
	` [None]: json`	

Invia l'immagine di backup ad Amazon S3.	In precedenza, un backup del database online veniva salvato nella directory `/backup` locale. Per inviare l'immagine di backup al bucket S3, esegui il seguente comando:
	`aws s3 sync /backup`
	` s3://logshipmig-db2/`
	`SAMPLE_backup`
		Amministratore AWS, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
Invia i log di archivio Db2 ad Amazon S3.	Sincronizza i log di archivio Db2 locali con il bucket S3 a cui può accedere l'istanza Db2 di destinazione su Amazon: EC2 `aws s3 sync /db2logs s3://logshipmig-db2/ SAMPLE_LOG` Esegui questo comando periodicamente utilizzando cron o altri strumenti di pianificazione. La frequenza dipende dalla frequenza con cui il database di origine archivia i file di registro delle transazioni.	Amministratore AWS, ingegnere addetto alla migrazione

Connetti Db2 su Amazon EC2 ad Amazon S3 e avvia la sincronizzazione del database

Operazione	Description	Competenze richieste
Crea un PKCS12 keystore.	Db2 utilizza un keystore di crittografia Public-Key Cryptography Standards (PKCS) per proteggere la chiave di accesso AWS. Crea un keystore e configura l'istanza Db2 di origine per utilizzarlo: `gsk8capicmd_64 -keydb -create -db "/home/db2inst1/.keystore/d`	DBA

Operazione	Description	Competenze richieste

	`b2s3.p12" -pw "<password>" -type pkcs12 -stash`
	
	`db2 "update dbm cfg using keystore_location /home/db2inst1/.keystore/db2s3.p12 keystore_type pkcs12"`	

Crea l'alias di accesso allo storage Db2.	Per creare l'alias di accesso allo storage, utilizzate la seguente sintassi dello script:
	
	`db2 "catalog storage access alias <alias_name> vendor S3 server <S3 endpoint> container '<bucket_name>'"`
	
	Ad esempio, lo script potrebbe avere il seguente aspetto:
	
	`db2 "catalog storage access alias DB2AWSS3 vendor S3 server s3.us-east-1.amazonaws.com container 'logshipmig-db2'"`
		DBA

Operazione	Description	Competenze richieste
Imposta l'area di staging.	Per impostazione predefinita, Db2 utilizza `DB2_OBJECT_STORAGE_LOCAL_STAGING_PATH` come area di gestione temporanea per caricare e scaricare file da e verso Amazon S3. Il percorso predefinito si trova nella home `sqllib/tmp/RemoteStorage.xxxx` directory dell'istanza, con `xxxx` riferimento al numero di partizione Db2. Si noti che l'area di gestione temporanea deve avere una capacità sufficiente per contenere le immagini di backup e i file di registro. È possibile utilizzare il registro per indirizzare l'area di gestione temporanea in una directory diversa. Consigliamo inoltre di utilizzare `DB2_ENABLE_COS_SDK=ON` e il collegamento alla `awssdk` libreria per bypassare l'area di staging di Amazon S3 per il backup e il ripristino del database: `DB2_OBJECT_STORAGE_SETTINGS=EnableStreamingRestore` `#By root:`	DBA

Operazione	Description	Competenze richieste

	`cp -rp /home/db2inst1/sqllib/lib64/awssdk/RHEL/7.6/* /home/db2inst1/sqllib/lib64/`
	
	`#By db2 instance owner:`
	`db2set DB2_OBJECT_STORAGE_LOCAL_STAGING_PATH=/db2stage`
	`db2set DB2_ENABLE_COS_SDK=ON`
	`Db2set DB2_OBJECT_STORAGE_SETTINGS=EnableStreamingRestore`
	`db2stop`
	`db2start`	

Ripristina il database dall'immagine di backup.	Ripristina il database di destinazione su Amazon EC2 dall'immagine di backup nel bucket S3: `db2 restore db sample from DB2REMOTE://DB2AWSS3/logshipmig-db2/SAMPLE_backup replace existing`	DBA

Operazione	Description	Competenze richieste

Esegui il rollforward del database.	Una volta completato il ripristino, il database di destinazione verrà messo in stato di attesa di rollforward. Configura `LOGARCHMETH1` e `LOGARCHMETH2` in modo che Db2 sappia dove trovare i file di registro delle transazioni:
	`db2 update db cfg for SAMPLE using LOGARCHMETH1 'DB2REMOTE://DB2AWSS3//SAMPLE_LOGS/'`
	`db2 update db cfg for SAMPLE using LOGARCHMETH2 OFF`
	Avvia il rollforward del database:
	`db2 ROLLFORWARD DATABASE sample to END OF LOGS`
	Questo comando elabora tutti i file di registro che sono stati trasferiti nel bucket S3. Eseguilo periodicamente in base alla frequenza del `s3 sync` comando sui server Db2 locali. Ad esempio, se `s3 sync` viene eseguito ogni ora e sono necessari 10 minuti per sincronizzare tutti i file di registro, imposta il comando in
		DBA

Operazione	Description	Competenze richieste
	modo che venga eseguito 10 minuti dopo ogni ora.	

Porta Db2 su Amazon EC2 online durante la finestra di cutover

Operazione	Description	Competenze richieste

Porta online il database di destinazione.	Durante la finestra di cutover, effettuate una delle seguenti operazioni:
	• Inserisci il database locale ed `ADMIN MODE` esegui il `s3 sync` comando per forzare l'archiviazione dell'ultimo registro delle transazioni.
	• Chiudi il database.
	
	Dopo la sincronizzazione dell'ultimo log delle transazioni in Amazon S3, esegui `ROLLFORWARD` il comando per l'ultima volta:
	
	`db2 rollforward DB sample to END OF LOGS`
	`db2 rollforward DB sample complete`
	
	`Rollforward Status`
	`....`
		DBA

Operazione	Description	Competenze richieste
	``` Rollforward status = not pending .... DB20000I The ROLLFORWA RD command completed successfully. db2 activate db sample DB20000I The ACTIVATE DATABASE command completed successfu lly. ``` Porta online il database di destinazione e indirizza le connessioni dell'applicazione a Db2 su Amazon EC2.	

risoluzione dei problemi

Problema	Soluzione
Se più database hanno lo stesso nome di istanza e lo stesso nome di database su host diversi (DEV, QA, PROD), i backup e i log potrebbero andare nella stessa sottodirectory.	Usa diversi bucket S3 per DEV, QA e PROD e aggiungi il nome host come prefisso della sottodirectory per evitare confusione.
Se sono presenti più immagini di backup nella stessa posizione, al momento del ripristino verrà visualizzato il seguente errore: `SQL2522N More than one backup file matches the time stamp value provided for the backed up database image.`	Nel `restore` comando, aggiungi il timestamp del backup: `db2 restore db sample from DB2REMOTE://DB2AWSS3/logshipmig-db2/SAMPLE_backup taken at 20230628164042 replace existing`

Risorse correlate

- Operazioni di backup e ripristino Db2 tra diversi sistemi operativi e piattaforme hardware
- Configura Db2 STORAGE ACCESS ALIAS e REMOTE DB2
- Comando Db2 ROLLFORWARD
- Metodo di archiviazione dei log secondario Db2

Esegui la migrazione di Db2 per LUW ad Amazon EC2 con disaster recovery ad alta disponibilità

Feng Cai, Aruna Gangireddy e Venkatesan Govindan, Amazon Web Services

Riepilogo

Quando i clienti migrano il loro carico di lavoro IBM Db2 LUW (Linux, UNIX e Windows) su Amazon Web Services (AWS), utilizzare Amazon Elastic Compute Cloud (Amazon EC2) con il modello Bring Your Own License (BYOL) è il modo più veloce. Tuttavia, la migrazione di grandi quantità di dati da Db2 locale ad AWS può essere una sfida, soprattutto quando la finestra di interruzione è breve. Molti clienti cercano di impostare la finestra di interruzione su meno di 30 minuti, il che lascia poco tempo per il database stesso.

Questo modello illustra come eseguire una migrazione Db2 con una breve finestra di interruzione utilizzando Db2 High Availability Disaster Recovery (HADR). Questo approccio si applica ai database Db2 che si trovano sulla piattaforma Linux little-endian e non utilizzano Data Partitioning Feature (DPF).

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un'istanza Db2 in esecuzione su un'istanza Amazon EC2 che corrisponde ai layout del file system locale
- Un bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) accessibile all'istanza EC2
- Una policy e un ruolo di AWS Identity and Access Management (IAM) per effettuare chiamate programmatiche ad Amazon S3
- Fuso orario e orologi di sistema sincronizzati su Amazon EC2 e sul server locale
- La rete locale connessa ad AWS tramite AWS Site-to-SiteVPN o AWS Direct Connect
- Comunicazione tra il server locale e Amazon EC2 su porte HADR

Limitazioni

- L'istanza locale Db2 e Amazon EC2 devono appartenere alla stessa famiglia di piattaforme.

- HADR non è supportato in un ambiente di database partizionato.
- HADR non supporta l'uso di raw I/O (accesso diretto al disco) per i file di registro del database.
- HADR non supporta la registrazione infinita.
- `LOGINDEXBUILD`deve essere impostato su`YES`, il che aumenterà l'utilizzo del registro per la ricostruzione dell'indice.
- Il carico di lavoro locale di Db2 deve essere registrato. Impostato `blocknonlogged=yes` nella configurazione del database per bloccare tutte le transazioni non registrate.

Versioni del prodotto

- Db2 per LUW versione 11.5.9 e successive

Architecture

Stack tecnologico di origine

- Db2 su Linux x86_64

Stack tecnologico Target

- Amazon EC2
- AWS Identity and Access Management (IAM)
- Simple Storage Service (Amazon S3)
- Site-to-SiteVPN AWS

Architettura Target

Nel diagramma seguente, Db2 locale viene eseguito `db2-server1` come principale. Ha due obiettivi di standby HADR. Un target di standby è in locale ed è opzionale. L'altro obiettivo in standby è Amazon EC2. `db2-ec2` Dopo che il database viene trasferito ad AWS, `db2-ec2` diventa il database principale.

1. I log vengono trasmessi in streaming dal database locale primario al database locale in standby.
2. Utilizzando Db2 HADR, i log vengono trasmessi in streaming dal database locale primario tramite Site-to-Site VPN a Db2 su Amazon EC2.

3. I log di backup e archiviazione di Db2 vengono inviati dal database locale primario al bucket S3 su AWS.

Tools (Strumenti)

Servizi AWS

- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.
- AWS Direct Connect collega la rete interna a una posizione Direct Connect tramite un cavo Ethernet standard in fibra ottica. Con questa connessione, puoi creare interfacce virtuali direttamente ai servizi AWS pubblici bypassando i provider di servizi Internet nel tuo percorso di rete.
- Amazon Elastic Compute Cloud (Amazon EC2) Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Site-to-Site VPN ti aiuta a trasferire il traffico tra le istanze che avvii su AWS e la tua rete remota.

Altri strumenti

- db2cli è il comando CLI interattivo di Db2.

Best practice

- Sul database di destinazione, utilizza gli endpoint gateway per Amazon S3 per accedere all'immagine di backup del database e ai file di log in Amazon S3.
- Sul database di origine, usa AWS PrivateLink per Amazon S3 per inviare l'immagine di backup del database e i file di log ad Amazon S3.

Epiche

Impostazione delle variabili di ambiente

Operazione	Description	Competenze richieste
Imposta le variabili di ambiente.	Questo modello utilizza i seguenti nomi e porte: 1. Nome host locale Db2: `db2-server1` 2. Nome host in standby HADR: `db2-server2` (se HADR è attualmente in esecuzione in locale) 3. Nome host Amazon EC2: `db2-ec2` 4. nome dell'istanza: `db2inst1` 5. Nome del database: `SAMPLE` 6. Porte HARD: • `db2-server1: 50010` • `db2-server2: 50011` • `db2-ec2: 50012` È possibile modificarle per adattarle al proprio ambiente.	DBA

Configura il server Db2 locale

Operazione	Description	Competenze richieste
Configura AWS CLI.	Per scaricare e installare la versione più recente di	Amministratore Linux

Operazione	Description	Competenze richieste

	AWS CLI, esegui i seguenti comandi:
	
	`$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"`
	`unzip awscliv2.zip`
	`sudo ./aws/install`	

Operazione	Description	Competenze richieste
Configura una destinazione locale per i log di archivio Db2.	Condizioni quali lavori di aggiornamento intensivi in batch e rallentamenti della rete possono causare un ritardo nel server di standby HADR. Per recuperare il ritardo, il server di standby necessita dei log delle transazioni del server primario. La sequenza di posizioni in cui richiedere i log è la seguente: • La directory di log attiva sul server primario • La `LOGARCHMETH2` posizione `LOGARCHMETH1` or sul server di standby • La `LOGARCHMETH2` posizione `LOGARCHMETH1` o sul server primario In questa configurazione, `/db2logs` viene impostata da `LOGARCHMETH2` on the source come area di staging. I log archiviati in questa directory verranno sincroniz zati in Amazon S3 e accessibi li da Db2 su Amazon EC2. Il modello utilizza `LOGARCHME TH2` because `LOGARCHME TH1` potrebbe essere stato configurato per utilizzare uno strumento di un fornitore di	DBA

Operazione	Description	Competenze richieste

	terze parti a cui il comando AWS CLI non può accedere:
	`db2 connect to sample`
	`db2 update db cfg for SAMPLE using LOGARCHMETH2 disk:/db2logs`	

Esegui un backup del database online.	Esegui un backup del database online e salvalo nel file system di backup locale: `db2 backup db sample online to /backup`	DBA

Configura il bucket S3 e la policy IAM

Operazione	Description	Competenze richieste
Crea un bucket S3.	Crea un bucket S3 per il server locale a cui inviare le immagini Db2 di backup e i file di log su AWS. Il bucket sarà accessibile da Amazon EC2: `aws s3api create-bucket --bucket hadrmig-db2 --region us-east-1`	Amministratore AWS
Creare una policy IAM	Il `db2bucket.json` file contiene la policy IAM per l'accesso al bucket S3: `{`	Amministratore AWS, amministratore di sistema AWS

Operazione	Description	Competenze richieste
	"Version": "2012-10-17", "Statement": [{ "Effect": "Allow", "Action": ["kms:GenerateDataKey", "kms:Decrypt", "s3:PutObject", "s3:GetObject", "s3:AbortMultipartUpload", "s3:ListBucket", "s3:DeleteObject", "s3:GetObjectVersion", "s3:ListMultipartUploadParts"], "Resource": ["arn:aws:s3:::hadrmig-db2/*", "arn:aws:s3:::hadrmig-db2"] }]	

Operazione	Description	Competenze richieste

	<pre>}</pre>
	Per creare la policy, usa il seguente comando AWS CLI:
	<pre>aws iam create-policy \
	      --policy-name
	 db2s3hapolicy \
	      --policy-document
	 file://db2bucket.j
	son</pre>
	L'output JSON mostra l'Amazon Resource Name (ARN) per la policy, `aws_account_id` dove rappresenta l'ID del tuo account:
	<pre>"Arn": "arn:aws:
	iam::aws_account_i
	d:policy/db2s3hapo
	licy"</pre>	

Operazione	Description	Competenze richieste
Allega la policy IAM al ruolo IAM.	Di solito, all'istanza EC2 con Db2 in esecuzione viene assegnato un ruolo IAM dall'amministratore di sistema. Se non viene assegnato alcun ruolo IAM, puoi scegliere Modifica ruolo IAM sulla console Amazon EC2. Collega la policy IAM al ruolo IAM associato all'istanza EC2. Dopo aver allegato la policy, l'istanza EC2 può accedere al bucket S3: `aws iam attach-role-policy --policy-arn "arn:aws:iam::aws_account_id:policy/db2s3hapolicy" --role-name db2s3harole`	

Inviare l'immagine di backup del database di origine e i file di log ad Amazon S3

Operazione	Description	Competenze richieste

Configura AWS CLI sul server Db2 locale.	Configura l'interfaccia a riga di comando di AWS con `Access Key ID` e `Secret Access Key` che hai generato in precedenza:
	
	`$ aws configure`
	`AWS Access Key ID [None]: **********`
		Amministratore AWS, amministratore di sistema AWS

Operazione	Description	Competenze richieste
	`AWS Secret Access Key [None]: ********* ********** Default region name [None]: us-east-1 Default output format [None]: json`	
Invia l'immagine di backup ad Amazon S3.	In precedenza, un backup del database online veniva salvato nella directory `/backup` locale. Per inviare l'immagine di backup al bucket S3, esegui il seguente comando: `aws s3 sync /backup s3://hadrmig-db2/SAMPLE_backup`	Amministratore AWS, amministratore di sistema AWS

Operazione	Description	Competenze richieste
Invia i log di archivio Db2 ad Amazon S3.	Sincronizza i log di archivio Db2 locali con il bucket Amazon S3 a cui può accedere l'istanza Db2 di destinazione su Amazon EC2: `aws s3 sync /db2logs s3://hadrmig-db2/SAMPLE_LOGS` Esegui questo comando periodicamente utilizzando cron o altri strumenti di pianificazione. La frequenza dipende dalla frequenza con cui il database di origine archivia i file di registro delle transazioni.	

Connetti Db2 su Amazon EC2 ad Amazon S3 e avvia la sincronizzazione iniziale del database

Operazione	Description	Competenze richieste
Crea un PKCS12 keystore.	Db2 utilizza un keystore di crittografia Public-Key Cryptography Standards (PKCS) per proteggere la chiave di accesso AWS. Crea un keystore e configura il Db2 sorgente per utilizzarlo: `gsk8capicmd_64 -keydb -create -db "/home/db2inst1/.keystore/db2s3.p12" -pw "<passwor`	DBA

Operazione	Description	Competenze richieste
	d>" -type pkcs12 - stash db2 "update dbm cfg using keystore_ location /home/db2 inst1/.keystore/db 2s3.p12 keystore_type pkcs12"	

Operazione	Description	Competenze richieste

Crea l'alias di accesso allo storage Db2.	Db2 utilizza un alias di accesso allo storage per accedere ad Amazon S3 direttamente con `INGEST` i comandi`LOAD`,`BACKUP DATABASE`, o. `RESTORE DATABASE`
	
	Perché hai assegnato un ruolo IAM all'istanza EC2 `USER` e `PASSWORD` non sei obbligato rio:
	
	`db2 "catalog storage access alias <alias_name> vendor S3 server <S3 endpoint> container '<bucket_name>'"`
	
	Ad esempio, lo script potrebbe avere il seguente aspetto:
	
	`db2 "catalog storage access alias DB2AWSS3 vendor S3 server s3.us-east-1.amazonaws.com container 'hadrmig-db2'"`
		DBA

Operazione	Description	Competenze richieste

Imposta l'area di staging.	Ti consigliamo di utilizzare `DB2_ENABLE_COS_SDK =ON` e il link alla `awssdk` libreria per bypassare l'area di staging di Amazon S3 per il backup e il ripristino del database: `DB2_OBJEC T_STORAGE_SETTINGS =EnableStreamingRe store`
	
	`#By root:`
	`cp -rp /home/db2inst1/ sqllib/lib64/awssdk/ RHEL/7.6/* /home/db2 inst1/sqllib/lib64/`
	
	`#By db2 instance owner:`
	`db2set DB2_OBJEC T_STORAGE_LOCAL_ST AGING_PATH=/db2stage`
	`db2set DB2_ENABL E_COS_SDK=ON`
	`db2set DB2_OBJEC T_STORAGE_LOCAL_ST AGING_PATH=/db2stage`
	`db2stop`
	`db2start`
		DBA

Operazione	Description	Competenze richieste

Ripristina il database dall'immagine di backup.	Ripristina il database di destinazione su Amazon EC2 dall'immagine di backup nel bucket S3:
	
	`db2 create db sample on /data1`
	`db2 restore db sample from DB2REMOTE://DB2AWSS3/hadrmig-db2/SAMPLE_backup replace existing`
		DBA

Configura HADR senza HADR in locale

Operazione	Description	Competenze richieste
Configura il server Db2 locale come principale.	Aggiorna le impostazioni di configurazione del database per HADR su `db2-server1` (l'origine locale) come principale. Imposta `HADR_SYNCMODE` sulla `SUPERASYNC` modalità, che ha il tempo di risposta delle transazioni più breve: `db2 update db cfg for sample using HADR_LOCAL_HOST db2-server1 HADR_LOCAL_SVC 50010 HADR_REMOTE_HOST db2-ec2 HADR_REMOTE_SVC 50012 HADR_REMOTE_INST db2inst1 HADR_SYNCMODE`	DBA

Operazione	Description	Competenze richieste
	`SUPERASYNC DB20000 I The UPDATE DATABASE CONFIGURATION command completed successfu lly` Sono previsti alcuni ritardi di rete tra il data center locale e AWS. (È possibile impostare un `HADR_SYNC MODE` valore diverso in base all'affidabilità della rete. Per ulteriori informazioni, vedere la sezione Risorse correlate).	
Cambia la destinazione dell'archivio dei log del database di destinazione.	Modifica la destinazione dell'archivio di log del database di destinazione in modo che corrisponda all'ambiente Amazon EC2: `db2 update db cfg for SAMPLE using LOGARCHME TH1 'DB2REMOTE://DB2AW SS3//SAMPLE_LOGS/' LOGARCHMETH2 OFF DB20000I The UPDATE DATABASE CONFIGURA TION command completed successfully`	DBA

Operazione	Description	Competenze richieste
Configura HADR per Db2 sul server Amazon EC2.	Aggiorna la configurazione del database per HADR in modalità standby: `db2-ec2` `db2 update db cfg for sample using HADR_LOCAL_HOST db2-ec2 HADR_LOCAL_SVC 50012 HADR_REMOTE_HOST db2-server1 HADR_REMOTE_SVC 50010 HADR_REMOTE_INST db2inst1 HADR_SYNCMODE SUPERASYNC DB20000I The UPDATE DATABASE CONFIGURATION command completed successfully`	DBA

Operazione	Description	Competenze richieste

Verifica la configurazione HADR.	Verificare i parametri HADR sui server Db2 di origine e di destinazione.
	
	Per verificare la configura zione su`db2-server1` , esegui il seguente comando:
	
	`db2 get db cfg for sample|grep HADR`
	`HADR database role = PRIMARY`
	`HADR local host name (HADR_LOCAL_HOST) = db2-server1`
	`HADR local service name (HADR_LOCAL_SVC) = 50010`
	`HADR remote host name (HADR_REMOTE_HOST) = db2-ec2`
	`HADR remote service name (HADR_REMOTE_SVC) = 50012`
	`HADR instance name of remote server (HADR_REMOTE_INST) = db2inst1`
	`HADR timeout value (HADR_TIMEOUT) = 120`
	`HADR target list (HADR_TARGET_LIST) =`
	`HADR log write synchronization mode`
		DBA

Operazione	Description	Competenze richieste
	(HADR_SYNCMODE) = NEARSYNC HADR spool log data limit (4KB) (HADR_SPOOL_LIMIT) = AUTOMATIC(52000) HADR log replay delay (seconds) (HADR_REPLAY_DELAY) = 0 HADR peer window duration (seconds) (HADR_PEER_WINDOW) = 0 HADR SSL certificate label (HADR_SSL_LABEL) = HADR SSL Hostname Validation (HADR_SSL_HOST_VAL) = OFF Per verificare la configurazione attivata`db2-ec2`, esegui il seguente comando: db2 get db cfg for sample|grep HADR HADR database role = STANDBY HADR local host name (HADR_LOCAL_HOST) = db2-ec2 HADR local service name (HADR_LOCAL_SVC) = 50012 HADR remote host name (HADR_REMOTE_HOST) = db2-server1	

Operazione	Description	Competenze richieste
	HADR remote service name (HADR_REMOTE_SVC) = 50010 HADR instance name of remote server (HADR_REMOTE_INST) = db2inst1 HADR timeout value (HADR_TIMEOUT) = 120 HADR target list (HADR_TARGET_LIST) = HADR log write synchronization mode (HADR_SYNCMODE) = SUPERASYNC HADR spool log data limit (4KB) (HADR_SPOOL_LIMIT) = AUTOMATIC(52000) HADR log replay delay (seconds) (HADR_REPLAY_DELAY) = 0 HADR peer window duration (seconds) (HADR_PEER_WINDOW) = 0 HADR SSL certificate label (HADR_SSL_LABEL) = HADR SSL Hostname Validation (HADR_SSL_HOST_VAL) = OFF I HADR_REMOTE_SVC parametri HADR_LOCAL_HOST HADR_LOCAL_SVC ,HADR_REMO	

Operazione	Description	Competenze richieste
	TE_H0ST , e indicano una configurazione HADR principale e una in standby.	

Avvia l'istanza Db2 HADR.	Avvia prima l'istanza Db2 HADR sul server di standby: db2-ec2
	
	`db2 start hadr on db sample as standby`
	`DB20000I The START HADR ON DATABASE command completed successfully.`
	
	Avvia Db2 HADR sul server primario (di origine): db2-server1
	
	`db2 start hadr on db sample as primary`
	`DB20000I The START HADR ON DATABASE command completed successfully.`
	
	La connessione HADR tra Db2 locale e Amazon EC2 è stata ora stabilita con successo. Il server primario Db2 db2-server1 avvia lo streaming dei record dei log delle transazioni in tempo reale. db2-ec2
		DBA

Configura HADR quando HADR esiste in locale

Operazione	Description	Competenze richieste
Aggiungi Db2 su Amazon EC2 come standby ausiliario.	Se HADR è in esecuzione sull'istanza Db2 locale, puoi aggiungere Db2 su Amazon EC2 come standby ausiliario eseguendo i seguenti comandi su: `HADR_TARGET_LIST db2-ec2` `db2 update db cfg for sample using HADR_LOCAL_HOST db2-ec2 HADR_LOCAL_SVC 50012 HADR_REMOTE_HOST db2-server1 HADR_REMOTE_SVC 50010 HADR_REMOTE_INST db2inst1 HADR_SYNCMODE SUPERASYNC DB20000I The UPDATE DATABASE CONFIGURATION command completed successfully. db2 update db cfg for sample using HADR_TARGET_LIST "db2-server1:50010|db2-server2:50011" DB20000I The UPDATE DATABASE CONFIGURATION command completed successfully.`	DBA

Operazione	Description	Competenze richieste
Aggiungi le informazioni di standby ausiliarie ai server locali.	Aggiornamento `HADR_TARGET_LIST` sui due server locali (primario e standby). Su `db2-server1`, esegui il codice seguente: `db2 update db cfg for sample using HADR_TARGET_LIST "db2-server2:50011|db2-ec2:50012" DB20000I The UPDATE DATABASE CONFIGURATION command completed successfully. SQL1363W One or more of the parameters submitted for immediate modification were not changed dynamically. For these configuration parameters, the database must be shutdown and reactivated before the configuration parameter changes become effective.`  `db2-server2` Attivo, esegui il codice seguente:  `db2 update db cfg for sample using HADR_TARGET_LIST "db2-serv`	DBA

Operazione	Description	Competenze richieste
	er1:50010|db2-ec2: 50012" DB20000I The UPDATE DATABASE CONFIGURATION command completed successfu lly. SQL1363W One or more of the parameter s submitted for immediate modificat ion were not changed dynamically. For these configura tion parameters, the database must be shutdown and reactivated before the configuration parameter changes become effective.	

Operazione	Description	Competenze richieste
Verifica la configurazione HADR.	Verificare i parametri HADR sui server Db2 di origine e di destinazione. Attivo`db2-server1` , esegui il codice seguente: `db2 get db cfg for sample|grep HADR HADR database role = PRIMARY HADR local host name (HADR_LOCAL_HOST) = db2-server1 HADR local service name (HADR_LOCAL_SVC) = 50010 HADR remote host name (HADR_REMOTE_HOST) = db2-server2 HADR remote service name (HADR_REMOTE_SVC) = 50011 HADR instance name of remote server (HADR_REMOTE_INST) = db2inst1 HADR timeout value (HADR_TIMEOUT) = 120 HADR target list (HADR_TARGET_LIST) = db2-server2:50011|db2-ec2:50012`	

Operazione	Description	Competenze richieste

	HADR log write synchronization mode (HADR_SYNCMODE) = NEARSYNC
	HADR spool log data limit (4KB) (HADR_SPOOL_LIMIT) = AUTOMATIC(52000)
	HADR log replay delay (seconds) (HADR_REPLAY_DELAY) = 0
	HADR peer window duration (seconds) (HADR_PEER_WINDOW) = 0
	HADR SSL certificate label (HADR_SSL_LABEL) =
	HADR SSL Hostname Validation (HADR_SSL_HOST_VAL) = OFF
	
	`db2-server2` Attivo, esegui il codice seguente:
	
	`db2 get db cfg for sample|grep HADR`
	`HADR database role = STANDBY`
	`HADR local host name (HADR_LOCAL_HOST) = db2-server2`
	`HADR local service name (HADR_LOCAL_SVC) = 50011`
	`HADR remote host name (HADR_REMOTE_HOST) = db2-server1`	

Operazione	Description	Competenze richieste
	HADR remote service name (HADR_REMOTE_SVC) = 50010 HADR instance name of remote server (HADR_REMOTE_INST) = db2inst1 HADR timeout value (HADR_TIMEOUT) = 120 HADR target list (HADR_TARGET_LIST) = db2-server1:50010|db2-ec2:50012 HADR log write synchronization mode (HADR_SYNCMODE) = NEARSYNC HADR spool log data limit (4KB) (HADR_SPOOL_LIMIT) = AUTOMATIC(52000) HADR log replay delay (seconds) (HADR_REPLAY_DELAY) = 0 HADR peer window duration (seconds) (HADR_PEER_WINDOW) = 0 HADR SSL certificate label (HADR_SSL_LABEL) = HADR SSL Hostname Validation (HADR_SSL_HOST_VAL) = OFF db2-ec2Attivo, esegui il codice seguente:	

Operazione	Description	Competenze richieste
	db2 get db cfg for sample|grep HADR HADR database role = STANDBY HADR local host name (HADR_LOC AL_HOST) = db2-ec2 HADR local service name (HADR_LOCAL_SVC) = 50012 HADR remote host name (HADR_REM OTE_HOST) = db2-serve r1 HADR remote service name (HADR_REMOTE_SVC) = 50010 HADR instance name of remote server (HADR_REMOTE_INST) = db2inst1 HADR timeout value (HADR_TIMEOUT) = 120 HADR target list (HADR_TAR GET_LIST) = db2-serve r1:50010|db2-serve r2:50011 HADR log write synchronization mode (HADR_SYNCMODE) = SUPERASYNC HADR spool log data limit (4KB) (HADR_SPOOL_LIMIT) = AUTOMATIC(52000)	

Operazione	Description	Competenze richieste
	HADR log replay delay (seconds) (HADR_REPLAY_DELAY) = 0 HADR peer window duration (seconds) (HADR_PEER_WINDOW) = 0 HADR SSL certificate label (HADR_SSL_LABEL) = HADR SSL Hostname Validation (HADR_SSL_HOST_VAL) = OFF I parametri HADR_TARGET_LIST HADR_LOCAL_HOST HADR_LOCAL_SVC ,HADR_REMOTE_HOST ,HADR_REMOTE_SVC , e indicano una configurazione HADR principale e due in standby.	

Operazione	Description	Competenze richieste

Arresta e avvia Db2 HADR.	`HADR_TARGET_LIST` è ora configurato su tutti e tre i server. Ogni server Db2 conosce gli altri due. Arresta e riavvia HADR (breve interruzi one) per sfruttare la nuova configurazione.
	
	Attivo`db2-server1` , esegui i seguenti comandi:
	
	`db2 stop hadr on db sample`
	`db2 deactivate db sample`
	`db2 activate db sample`
	
	`db2-server2` Attivo, esegui i seguenti comandi:
	
	`db2 deactivate db sample`
	`db2 start hadr on db sample as standby`
	`SQL1766W The command completed successfully`
	
	`db2-ec2`Attivo, esegui i seguenti comandi:
	
	`db2 start hadr on db sample as standby`
	`SQL1766W The command completed successfully`
	
	`db2-server1` Attivo, esegui i seguenti comandi:
		DBA

Operazione	Description	Competenze richieste

	`db2 start hadr on db sample as primary`
	`SQL1766W The command completed successfully`
	
	La connessione HADR tra Db2 locale e Amazon EC2 è ora stabilita con successo. Il server primario Db2 `db2-server1` avvia lo streaming dei record dei log delle transazioni su entrambi `db2-server2` e `db2-ec2` in tempo reale.	

Imposta Db2 su Amazon EC2 come principale durante la finestra di cutover

Operazione	Description	Competenze richieste

Verificare che non vi sia alcun ritardo HADR sul server di standby.	Controllate lo stato HADR dal server primario. `db2-server1` Non allarmate vi quando `HADR_STATE` è in `REMOTE_CATCHUP` stato, il che è normale quando `HADR_SYNCMODE` è impostato su. `SUPERASYNC` Poi `PRIMARY_LOG_TIME` e `STANDBY_REPLAY_LOG_TIME` mostrano che sono sincronizzati:
	
	`db2pd -hadr -db sample`
	
	`HADR_ROLE = PRIMARY`
		DBA

Operazione	Description	Competenze richieste
	REPLAY_TYPE = PHYSICAL HADR_SYNCMODE = SUPERASYNC STANDBY_ID = 2 LOG_STREAM_ID = 0 HADR_STATE = REMOTE_CATCHUP ..... PRIMARY_LOG_TIME = 10/26/2022 02:11:32.000000 (1666750292) STANDBY_LOG_TIME = 10/26/2022 02:11:32.000000 (1666750292) STANDBY_REPLAY_LOG_TIME = 10/26/2022 02:11:32.000000 (1666750292)	

Operazione	Description	Competenze richieste

Esegui HADR Takeover.	Per completare la migrazione, crea `db2-ec2` il database primario eseguendo il comando HADR takeover. Utilizzate il comando `db2pd` per verificare il `HADR_ROLE` valore:
	
	`db2 TAKEOVER HADR ON DATABASE sample`
	`DB20000I The TAKEOVER HADR ON DATABASE command completed successfully.`
	
	`db2pd -hadr -db sample`
	`Database Member 0 -- Database SAMPLE -- Active -- Up 0 days 00:03:25 -- Date 2022-10-26-02.46.45.048988`
	
	`HADR_ROLE = PRIMARY`
	
	`REPLAY_TYPE = PHYSICAL`
	
	Per completare la migrazione verso AWS, indirizza le connessioni dell'applicazione a Db2 su Amazon EC2.	

Risoluzione dei problemi

Problema	Soluzione
Se utilizzi NAT per motivi di firewall e sicurezza , l'host può avere due indirizzi IP (uno interno e uno esterno), il che può causare un errore di controllo dell'indirizzo IP HADR. Il `START HADR ON DATABASE` comando restituirà il seguente messaggio: `HADR_LOCAL_HOST:HADR_LOCAL_SVC (-xx-xx-xx-xx.:50011 (xx.xx.xx .xx:50011)) on remote database is different from HADR_REMOTE_HOST:H ADR_REMOTE_SVC (xx-xx-xx- xx.:50011 (x.x.x.x:50011)) on local database.`	Per supportare HADR in un ambiente NAT, è possibile configurarlo `HADR_LOCAL_HOST` con l'indirizzo interno ed esterno. Ad esempio, se il server Db2 ha il nome interno `host1` e il nome esterno`host1E`, `HADR_LOCAL_HOST` può essere. `HADR_LOCAL_HOST: "host1 | host1E"`

Risorse correlate

- Operazioni di backup e ripristino Db2 tra diversi sistemi operativi e piattaforme hardware
- Configura Db2 STORAGE ACCESS ALIAS e REMOTE DB2
- Disaster recovery ad alta disponibilità Db2
- hadr_syncmode - Modalità di sincronizzazione HADR per le scritture di log nel parametro di configurazione dello stato peer

Esegui la migrazione di applicazioni ospitate da IIS su Amazon utilizzando appcmd.exe EC2

Deepak Kumar, Amazon Web Services

Riepilogo

Quando esegui la migrazione di applicazioni ospitate da Internet Information Services (IIS) su istanze Amazon Elastic Compute Cloud (Amazon EC2), devi affrontare diverse sfide di autenticazione. Queste sfide includono il reinserimento delle credenziali di dominio per le identità dei pool di applicazioni e la potenziale rigenerazione delle chiavi macchina per la corretta funzionalità del sito web. È possibile utilizzarlo AWS Directory Service per stabilire relazioni di fiducia con Active Directory locale o creare una nuova Active Directory gestita in. AWS Questo modello descrive un approccio di migrazione pulito che utilizza la funzionalità di backup e ripristino di IIS su EC2 istanze Amazon. L'approccio utilizza appcmd.exe per disinstallare e reinstallare IIS sulle EC2 istanze di destinazione, permettendo una corretta migrazione dei siti Web ospitati su IIS, delle identità dei pool di applicazioni e delle chiavi di macchina.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS per il server di destinazione.
- Un server IIS di origine funzionale con siti Web ospitati su di esso.
- Comprensione dei principi di funzionamento di IIS come amministrazione e configurazione.
- Accesso da amministratore di sistema sia sul server di origine che su quello di destinazione.
- Migrazione completata del server IIS di origine verso la destinazione Account AWS. Puoi utilizzare strumenti di migrazione come AWS Application Migration Service un approccio basato su snapshot di Amazon Machine Image (AMI) o altri strumenti di migrazione.

Limitazioni

- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per informazioni sulla disponibilità per regione, consulta AWS Servizi per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- IIS 8.5 o IIS 10.0

Architecture

Stack di tecnologia di origine

- Windows Server con IIS 8.5 o IIS 10.0 installato

Stack tecnologico Target

- Windows Server con IIS 8.5 o IIS 10.0 installato
- Application Migration Service

Architettura di destinazione

Il diagramma seguente mostra i componenti del flusso di lavoro e dell'architettura per questo modello.

La soluzione include i seguenti passaggi:

1. Installa e configura l'agente di AWS replica sul server IIS di origine nel data center aziendale. Questo agente avvia il processo di replica e gestisce il trasferimento dei dati verso. AWS
2. L'agente di AWS replica stabilisce una connessione sicura all'Application Migration Service e inizia a replicare i dati del server di origine, incluse le configurazioni IIS, i siti Web e i file delle applicazioni.
3. Application Migration Service avvia EC2 le istanze nella sottorete dell'applicazione con i dati replicati. L' EC2 istanza di destinazione esegue IIS e contiene le applicazioni migrate con i volumi Amazon Elastic Block Store (Amazon EBS) associati. Dopo la replica iniziale, Application Migration Service continua a sincronizzare le modifiche finché non sei pronto per passare al nuovo ambiente.

Tools (Strumenti)

Servizi AWS

- AWS Application Migration Serviceti aiuta a riospitare (lift and shift) le applicazioni Cloud AWS senza modifiche e con tempi di inattività minimi.

- Amazon Elastic Block Store (Amazon EBS) fornisce volumi di storage a livello di blocco da utilizzare con le istanze Amazon. EC2
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.

Altri strumenti

- Internet Information Services (IIS) per Windows Server è un server Web con un'architettura scalabile e aperta per l'hosting di qualsiasi cosa sul Web. IIS offre una serie di strumenti di amministrazione, tra cui strumenti di amministrazione e da riga di comando (ad esempio, appcmd.exe), codice e script gestiti e APIs supporto per Windows PowerShell .

Epiche

Esegui il backup di IIS all'origine prima della migrazione

Operazione	Description	Competenze richieste
Crea backup di siti Web ospitati su IIS, chiave di configurazione e chiave. `WAS`	Per creare backup per siti Web ospitati su IIS, la chiave di configurazione (`iisConfigurationKey`) e la `WAS` chiave (`iisWasKey`), utilizzate appcmd.exe sul server di origine. Utilizza le fasi seguenti: 1. Apri un prompt dei comandi ed eseguilo come amministratore. 2. Esegui il comando riportato qui di seguito: `c:\windows\system32\inetsrv` 3. Esegui il comando seguente, sostituendolo	Amministratore IIS

Operazione	Description	Competenze richieste
	<backupname> con il tuo valore: `appcmd.exe add backup "<backupname>"` Per esportare la chiave di configurazione e la `WAS` chiave, procedi come segue: 1. Esegui il comando riportato qui di seguito: `cd c:\windows\Microsoft.NET\Framework\v2.0.50727 5. aspnet_regiis.exe -px "iisConfigurationKey"` 2. Esegui il comando riportato qui di seguito: `C:\temp\iisConfKey.xml -pri` 3. Esegui il comando riportato qui di seguito: `aspnet_regiis.exe -px "iisWasKey" C:\temp\iisWasKey.xml -pri`	

Disinstalla e reinstalla IIS sul server di destinazione

Operazione	Description	Competenze richieste
Disinstalla IIS sul server di destinazione.	Per disinstallare IIS sul server di destinazione, attenersi alla seguente procedura:	Amministratore IIS

Operazione	Description	Competenze richieste
	1. Accedi al server con diritti di amministratore. 2. Vai a Server manager. 3. Scegli Gestisci, Rimuovi ruoli e funzionalità, Avanti e Avanti. 4. Deseleziona la casella di controllo Web IIS, quindi scegli Avanti, Avanti. 5. Seleziona la casella di controllo Riavvia automatic amente il server di destinazi one. Una volta completat a la disinstallazione, il sistema si riavvia. 6. Verifica che IIS sia stato disinstallato correttamente accedendo a `C:\Windows\System32\inetsrv` e confermando che questa cartella è vuota.	

Operazione	Description	Competenze richieste
Installa IIS sul server di destinazione.	Per installare IIS sul server di destinazione, attenersi alla seguente procedura: 1. Accedi al server con diritti di amministratore. 2. Vai a Server manager. 3. Scegli Aggiungi ruoli e funzionalità, Avanti, Avanti, Avanti e Avanti. 4. Seleziona la casella di controllo Webserver IIS. Quando richiesto, scegli Aggiungi funzionalità. 5. Scegli tutte le caselle di controllo mostrate in Sviluppo dell'app, Aggiungi funzionalità, Seleziona tutte le funzionalità. Puoi selezionare le funzionalità a livello granulare appena prima della conferma. 6. Scegli Installa.	Amministratore IIS

Ripristina i siti Web e la configurazione IIS dai backup

Operazione	Description	Competenze richieste
Ripristina i siti Web e la configurazione IIS.	Per ripristinare i backup IIS creati dal server di origine sul server di destinazione, attenersi alla seguente procedura:	Amministratore IIS

Operazione	Description	Competenze richieste
	1. Crea una cartella denominata `backup` nella `C:\windows\system32\inetsrv` cartella e inserisci la cartella di backup di origine al suo interno. 2. Apri un prompt dei comandi ed eseguilo come amministratore. 3. Esegui il comando riportato qui di seguito: `cd C:\windows\system32\inetsrv` 4. Esegui il comando riportato qui di seguito: `C:\windows\system32\inetsrv\appcmd.exe list backup` 5. Esegui il comando riportato qui di seguito: `C:\windows\system32\inetsrv\appcmd.exe restore backup <sourcebackup>` 6. Esegui il comando riportato qui di seguito: `cd c:\windows\Microsoft.NET\Framework64\v2.0.50727\` 7. Esegui il comando riportato qui di seguito: `aspnet_regiis.exe -pi`	

Operazione	Description	Competenze richieste
	`"iisConfigurationKey" C:\temp\iisConfKey.xml -exp` 8. Esegui il comando riportato qui di seguito: `aspnet_regiis.exe -pi "iisWasKey" C:\temp\iisWasKey.xml -exp` 9. Reimposta IIS e vai alle impostazioni avanzate di Application Pool Identity e Website per reinserire la password.	

Risorse correlate

AWS documentazione

- Installazione del AWS Replication Agent (AWS Application Migration Service documentazione)

AWS Linee guida prescrittive

- Esegui la migrazione di una macchina virtuale locale su Amazon utilizzando EC2 AWS Application Migration Service
- Utilizzo delle AMIs nostre istantanee di Amazon EBS per i backup

Risorse Microsoft

- Identità del pool di applicazioni
- Documentazione IIS
- Documentazione IIS 8 appcmd.exe
- Documentazione IIS 10 appcmd.exe

- Potenti strumenti di amministrazione

Esegui la migrazione di un database Microsoft SQL Server locale su Amazon EC2 utilizzando Application Migration Service

Senthil Ramasamy, Amazon Web Services

Riepilogo

Questo modello descrive i passaggi per la migrazione di un database Microsoft SQL Server da un data center locale a un'istanza Amazon Elastic Compute Cloud (Amazon EC2). Utilizza AWS Application Migration Service (AWS MGN) per riospitare il database utilizzando una migrazione automatizzata. lift-and-shift AWS MGN esegue la replica a livello di blocco del server del database di origine.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un database Microsoft SQL Server di origine in un data center locale

Limitazioni

- La larghezza di banda della rete potrebbe essere limitata tra il data center locale e. AWS
- AWS MGN è limitato ai database ospitati su server autonomi con archiviazione dedicata. Non supporta la migrazione di sistemi di database in cluster e di sistemi di database in cui la velocità di modifica supera il throughput di rete.
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- Tutte le versioni del database Microsoft SQL Server
- Sistemi operativi Windows e Linux che supportano AWS MGN

Architecture

Stack tecnologico di origine

Un database Microsoft SQL Server locale

Stack tecnologico Target

Un database Microsoft SQL Server su un' EC2 istanza Amazon

Architettura Target

Questa architettura utilizza AWS MGN per replicare i dati da un data center aziendale locale a. AWS Il diagramma mostra il processo di replica dei dati, le comunicazioni tramite API e le fasi di test e cutover.

1. Replica dei dati:
   - AWS MGN replica i dati dal data center aziendale locale AWS e avvia la replica continua delle modifiche.
   - I server di replica nella sottorete di staging ricevono ed elaborano i dati.
2. Comunicazione tramite API:
   - I server di replica si connettono agli AWS endpoint delle API MGN EC2, Amazon e Amazon Simple Storage Service (Amazon S3) tramite la porta TCP 443.
   - AWS MGN gestisce la migrazione.
   - Amazon EC2 gestisce le operazioni delle istanze.
3. Test e cutover:
   - Le istanze di test vengono avviate nella sottorete operativa utilizzando dati replicati.
   - Dopo il successo dei test, AWS MGN crea istanze cutover per la migrazione finale.

Tools (Strumenti)

- AWS Application Migration Service (AWS MGN) vi aiuta a riospitare (lift and shift) le applicazioni Cloud AWS senza modifiche e con tempi di inattività minimi.
- Direct Connectcollega la rete interna a una posizione Direct Connect tramite un cavo Ethernet standard in fibra ottica. Con questa connessione, è possibile creare interfacce virtuali direttamente ai AWS servizi pubblici ignorando i provider di servizi Internet nel percorso di rete.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.

- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Best practice

- Configura gli endpoint regionali delle API per AWS MGN EC2, Amazon e Amazon S3 nel cloud privato virtuale (VPC) per vietare l'accesso pubblico da Internet.
- Configura le impostazioni di avvio di AWS MGN per avviare i server di database di destinazione in una sottorete privata.
- Consenti solo le porte necessarie nei gruppi di sicurezza del database.
- Segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.

Epiche

Configura AWS MGN

Operazione	Description	Competenze richieste
Configura AWS MGN.	Cercate il file AWS Application Migration Service nel Console di gestione AWS e avviate il processo di configurazione. Questo creerà un modello di replica e ti reindirizzerà alla pagina dei server di origine della console MGN. Durante la configurazione del servizio MGN, scegliete un ruolo di servizio dall'elenco generato.	DBA, ingegnere addetto alla migrazione
Aggiungi server di origine.	Aggiungi i dettagli del server di database di origine locale, quindi aggiungi il server.	DBA, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
Installa l'agente AWS MGN sul server di origine.	Scarica il programma di installazione dell'agente AWS MGN sul tuo sistema locale e trasferiscilo sul server del database di origine. Per convalidare l'hash del programma di installazione, vedere Convalida del programma di installazione di Replication Agent scaricato AWS per Windows 2012.	DBA, ingegnere addetto alla migrazione

Installa l'agente AWS MGN sui computer di origine

Operazione	Description	Competenze richieste
Genera credenziali IAM del client.	Prima di installare l'agente AWS MGN, genera AWS le credenziali creando un nuovo utente IAM con le autorizzazioni appropriate. Per ulteriori informazioni, consulta le policy gestite da AWS AWS Application Migration Service e la generazione delle AWS credenziali richieste.	DBA, ingegnere addetto alla migrazione
Installa l'agente sul server di origine.	Installare l'agente sul computer di origine che ospita il database Microsoft SQL Server. Per ulteriori informazioni, vedere Installazione	Proprietario dell'app, DBA, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	dell'agente di AWS replica sui server Windows. Fornisci le seguenti credenziali AWS: • Regione AWS • AWS ID della chiave di accesso • AWS chiave di accesso segreta Le tue AWS credenziali uniche consentono all'agente AWS MGN di autenticare ed eseguire attività di migrazione.	

Operazione	Description	Competenze richieste
Scegli i dischi da replicare.	Dopo aver inserito AWS le credenziali, il programma di installazione verifica che il server soddisfi i requisiti minimi per l'installazione dell'agente (ad esempio, se il server dispone di spazio su disco sufficiente per installar e l'agente MGN). AWS Il programma di installazione visualizza le etichette dei volumi e i dettagli di archiviaz ione. Per replicare il database utilizzando il servizio AWS MGN, selezionate i dischi applicabili sul server di origine. Immettete il percorso di ogni disco, separato da virgole. Se desideri replicare tutti i dischi, lascia il percorso vuoto. Dopo aver confermato i dischi selezionati, l'installazione procede.	DBA, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
Monitora l'avanzamento della sincronizzazione.	AWS Replication Agent avvia il processo di sincronizzazione acquisendo prima un'istant anea dei dischi selezionati e quindi replicando i dati. È possibile monitorare l'avanzamento della sincroniz zazione dalla pagina del server di origine nella console MGN. AWS Per ulteriori informazioni, consultate Monitoraggio del server nel ciclo di vita della migrazione.	DBA, ingegnere addetto alla migrazione

Replica con AWS MGN

Operazione	Description	Competenze richieste
Gestisci l'avanzamento della replica.	Dopo aver avviato la sincroniz zazione iniziale, il server di origine viene visualizzato nella console AWS MGN, dove è possibile gestire e monitorar e la migrazione. La console visualizza il tempo stimato per la replica completa, basato sulla dimensione totale dei dischi selezionati e sulla larghezza di banda di rete disponibile.	DBA, ingegnere addetto alla migrazione
Verifica la sincronizzazione.	Dopo la completa sincroniz zazione dei dischi sul server di origine, verifica che tutti	Proprietario dell'app, DBA, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	i dischi selezionati siano elencati come completam ente sincronizzati e che non vengano segnalati errori nella console. La console AWS MGN trasferir à quindi automaticamente lo stato del ciclo di vita della migrazione a Pronto per il test, a indicare che l'ambiente replicato in AWS è pronto per il test di prestazioni e funzional ità.	

Prova e taglia

Operazione	Description	Competenze richieste
Configura le impostazioni di avvio.	Scegliete il server di origine nella console AWS MGN e aggiornate le impostazioni di avvio per l'istanza di test di destinazione. Dalla pagina dei dettagli del server di origine, accedete alla scheda Impostazioni di avvio per configurare l'istanza di test. Scegli un tipo di istanza conveniente e un tipo di volume Amazon Elastic Block Store (Amazon EBS), quindi configura i gruppi di sicurezza e i requisiti di rete. Per ulteriori	DBA, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	informazioni, consulta le impostazioni di Launch.	
Avvia l'istanza di test di destinazione.	Passa alla console AWS MGN del tuo computer di origine sincronizzato e avvia un'istanza di test di destinazione scegliendo Test and cut over e quindi Launch test instances. Questo crea un job di avvio che distribuisce l'istanza di test utilizzando le impostazioni configurate. L'istanza viene avviata Cloud AWS e replica l'ambiente del server del database di origine. Monitora l'avanzamento del lancio dalla pagina Cronologia dei lanci, dove puoi tenere traccia della creazione dell'istanza e risolvere eventuali problemi.	DBA, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
Convalida l'istanza di test di destinazione.	Convalida il server di EC2 database Amazon: 1. Assicurati che la console AWS MGN mostri che l'istanza di test funziona correttamente. 2. Utilizzate un client RDP per accedere. 3. Dal menu Start, apri SQL Server Configuration Manager. 4. Verifica lo stato dei servizi SQL Server, inclusi SQL Server (`MSSQLSERVER`) e SQL Server Agent, assicurandoti che siano intatti e impostati sul tipo di avvio corretto. 5. Confronta le impostazioni e le configurazioni del disco tra l'istanza di test e il server del database di origine, confermando che le lettere di unità, i volumi, le allocazioni dei dischi e le directory necessarie siano mappate correttamente. 6. Connettiti a SQL Server sull' EC2 istanza Amazon di prova e verifica che tutti i database di origine siano migrati e siano presenti nell'elenco dei database.	DBA, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	Esegui test di convalida per garantire che il database funzioni come previsto.	

Operazione	Description	Competenze richieste

Rinomina il server.	AWS La migrazione a MGN prevede una copia a livello di storage del server di origine locale. L' EC2 istanza di SQL Server contiene solo i dettagli del server di origine originale nei file binari, pertanto aggiorna le informazioni binarie in modo che riflettano il nome del nuovo server.
	1. Usa SQL Server Management Studio (SSMS) per connetterti alla tua istanza di SQL Server EC2 .
	2. Controlla il nome del server:
	`SELECT @@SERVERNAME AS 'Current Server Name';`
	3. Rinomina l'istanza di SQL Server sostituendola `NEW_SERVER_NAME` con il nome del tuo server:
	`EXEC sp_dropserver 'OLD_SERVER_NAME'`
	`EXEC sp_addserver 'NEW_SERVER_NAME', 'local';`
	4. Verifica che il nome del server sia corretto:
		DBA, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	`SELECT @@SERVERNAME AS 'Updated Server Name';` 5. Riavviare l'istanza di SQL Server.	

Operazione	Description	Competenze richieste
Avvia l'istanza cutover.	Nella console AWS MGN, nella pagina Server di origine, confermate che lo stato del ciclo di vita della migrazion e del server sia Pronto per il cutover. Configura le impostazioni di avvio per l'istanza cutover, assicuran doti che rispecchino il tuo ambiente locale. Prima di iniziare il cutover, chiudi il database locale, per garantire quanto segue: • Tutte le transazioni in corso sono complete. • Non si verificheranno nuove transazioni durante il processo di cutover. • La sincronizzazione dei dati tra i dischi di origine e di destinazione è completa. Avviate l'istanza cutover nella console MGN. AWS Quando l'istanza cutover è operativa , accedete all'istanza ed eseguite i seguenti test: 1. Assicurati che SQL Server si avvii correttamente e che i database siano accessibili.	Proprietario dell'app, DBA, ingegnere addetto alla migrazione, responsabile della migrazione

Operazione	Description	Competenze richieste
	2. Verifica che i tuoi dati siano intatti e coerenti con il server di origine. 3. Esegui tutti i test delle applicazioni per confermar e che funzionino come previsto. 4. Nella console AWS MGN, imposta lo stato di migrazione su Cutover complete. 5. Iniziate a indirizzare il traffico verso l'istanza. EC2	

risoluzione dei problemi

Problema	Soluzione
La sincronizzazione iniziale fallisce nella fase di autenticazione.	Si tratta di un problema di connettività di rete. Il server di replica non può connettersi a AWS MGN.

Risorse correlate

Documentazione AWS

- Guida introduttiva con AWS Application Migration Service
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon EC2
- Cos'è Microsoft SQL Server su Amazon EC2?

Video

- Esecuzione di una migrazione Lift and Shift con AWS Application Migration Service (video)

Esegui la migrazione di un carico di lavoro F5 BIG-IP su F5 BIG-IP VE su Cloud AWS

Suresh Veeragoni e Deepak Kumar, Amazon Web Services

Riepilogo

Le organizzazioni stanno cercando di migrare verso il per Cloud AWS aumentare la loro agilità e resilienza. Dopo aver migrato le tue soluzioni di sicurezza e gestione del traffico F5 BIG-IP verso Cloud AWS, puoi concentrarti sull'agilità e sull'adozione di modelli operativi di alto valore nell'architettura aziendale.

Questo modello descrive come migrare un carico di lavoro F5 BIG-IP verso un carico di lavoro F5 BIG-IP Virtual Edition (VE) su. Cloud AWS Il carico di lavoro verrà migrato mediante il rehosting dell'ambiente esistente e l'implementazione di aspetti del replatforming, come l'individuazione dei servizi e le integrazioni delle API. AWS CloudFormation i modelli accelerano la migrazione del carico di lavoro verso. Cloud AWS

Questo modello è destinato ai team di ingegneria tecnica e architettura che stanno migrando le soluzioni di sicurezza e gestione del traffico di F5 e accompagna la guida Migrating from F5 BIG-IP to F5 BIG-IP VE sul sito web on the Prescriptive Guidance. Cloud AWS AWS

Prerequisiti e limitazioni

Prerequisiti

- Un carico di lavoro F5 BIG-IP locale esistente.
- Licenze F5 esistenti per le versioni BIG-IP VE.
- Account AWS Un attivo.
- Un cloud privato virtuale (VPC) esistente configurato con uscita tramite un gateway NAT o un indirizzo IP elastico e configurato con accesso ai seguenti endpoint: Amazon Simple Storage Service (Amazon S3), Amazon Elastic Compute Cloud (Amazon), () e Amazon. EC2 AWS Security Token Service AWS STS CloudWatch Puoi anche modificare l'architettura VPC modulare e scalabile Quick Start come elemento costitutivo per le tue implementazioni.
- Una o due zone di disponibilità esistenti, a seconda delle esigenze.
- Tre sottoreti private esistenti in ogni zona di disponibilità.
- AWS CloudFormation modelli, disponibili nel repository GitHub F5.

Durante la migrazione, puoi anche utilizzare quanto segue, a seconda delle tue esigenze:

- Un'estensione F5 Cloud Failover per gestire la mappatura elastica degli indirizzi IP, la mappatura IP secondaria e le modifiche alla tabella di routing.
- Se utilizzi più zone di disponibilità, dovrai utilizzare le F5 Cloud Failover Extensions per gestire la mappatura elastica degli IP sui server virtuali.
- È consigliabile prendere in considerazione l'utilizzo di F5 Application Services 3 (AS3), F5 Application Services Templates (FAST) o un altro modello di infrastruttura come codice (IaC) per gestire le configurazioni. La preparazione delle configurazioni in un modello IAc e l'utilizzo di repository di codice contribuiranno alla migrazione e alle attività di gestione continue.

Competenza

- Questo modello richiede familiarità con il modo in cui uno o più centri dati VPCs possono essere collegati ai data center esistenti. Per ulteriori informazioni su questo argomento, consulta le opzioni di connettività Network-to-Amazon VPC nella documentazione di Amazon VPC.
- È inoltre richiesta familiarità con i prodotti e i moduli F5, tra cui Traffic Management Operating System (TMOS), Local TrafficManager (LTM), Global Traffic Manager (GTM), Access Policy Manager (APM), Application Security Manager (ASM), Advanced Firewall Manager (AFM) e BIG-IQ.

Versioni del prodotto

- Si consiglia di utilizzare F5 BIG-IP versione 13.1 o successiva, sebbene il modello supporti F5 BIG-IP versione 12.1 o successiva.

Architecture

Stack tecnologico di origine

- Carico di lavoro F5 BIG-IP

Stack tecnologico Target

- Amazon CloudFront
- CloudWatch
- Amazon EC2
- Simple Storage Service (Amazon S3)

- Amazon VPC
- AWS Global Accelerator
- AWS STS
- AWS Transit Gateway
- F5 BIG-IP VE

Architettura Target

Tools (Strumenti)

- CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su e. Account AWS Regioni AWS
- Amazon CloudFront accelera la distribuzione dei tuoi contenuti web distribuendoli attraverso una rete mondiale di data center, che riduce la latenza e migliora le prestazioni.
- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue AWS risorse e delle applicazioni su cui esegui AWS in tempo reale.
- Amazon Elastic Compute Cloud (Amazon EC2) offre capacità di elaborazione scalabile in Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e scalarli rapidamente verso l'alto o verso il basso.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Security Token Service (AWS STS) consente di richiedere credenziali temporanee con privilegi limitati per gli utenti.
- AWS Transit Gatewayè un hub centrale che collega cloud privati virtuali (VPCs) e reti locali.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.

Epiche

Scoperta e valutazione

Operazione	Description	Competenze richieste
Valuta le prestazioni di F5 BIG-IP.	Raccogli e registra le metriche delle prestazioni delle applicazioni sul server virtuale e le metriche dei sistemi che verranno migrati. Ciò contribui rà a dimensionare correttam ente l' AWS infrastruttura di destinazione per una migliore ottimizzazione dei costi.	Architetto F5, ingegnere e architetto di rete, ingegnere
Valuta il sistema operativo e la configurazione di F5 BIG-IP.	Valuta quali oggetti verranno migrati e se è necessario mantenere una struttura di rete, ad esempio. VLANs	Architetto F5, ingegnere
Valuta le opzioni di licenza F5.	Valuta la licenza e il modello di consumo necessari. Questa valutazione deve essere basata sulla valutazione del sistema operativo e della configurazione di F5 BIG-IP.	Architetto F5, ingegnere
Valuta le applicazioni pubbliche.	Determina quali applicazi oni richiederanno indirizzi IP pubblici. Allinea tali applicazi oni alle istanze e ai cluster richiesti per soddisfare i requisiti relativi alle prestazio ni e agli accordi sul livello di servizio (SLA).	Architetto F5, architetto del cloud, architetto di rete, ingegnere, team di app

Operazione	Description	Competenze richieste
Valuta le applicazioni interne.	Valuta quali applicazioni verranno utilizzate dagli utenti interni. Assicurati di sapere dove si trovano gli utenti interni all'interno dell'orga nizzazione e in che modo tali ambienti si connetton o a Cloud AWS. È inoltre necessario assicurarsi che tali applicazioni possano utilizzar e il Domain Name System (DNS) come parte del dominio predefinito.	F5 Architect, Cloud Architect , Network Architect, Engineer, App Teams
Finalizza l'AMI.	Non tutte le versioni F5 BIG-IP vengono create come Amazon Machine Images (). AMIs Puoi utilizzare lo strumento F5 BIG-IP Image Generator se disponi di versioni specifiche richieste di Quick-Fix Engineering (QFE). Per ulteriori informazi oni su questo strumento, consultate la sezione «Risorse correlate».	Architetto F5, architetto cloud, ingegnere
Finalizza i tipi e l'architettura delle istanze.	Decidi i tipi di istanza, l'archite ttura VPC e l'architettura interconnessa.	Architetto F5, architetto del cloud, architetto di rete, ingegnere

Attività complete relative alla sicurezza e alla conformità

Operazione	Description	Competenze richieste
Documenta le politiche di sicurezza esistenti di F5.	Raccogli e documenta le politiche di sicurezza F5 esistenti. Assicurati di crearne una copia in un repository di codice sicuro.	Architetto F5, ingegnere
Crittografa l'AMI.	(Facoltativo) L'organizzazione potrebbe richiedere la crittografia dei dati inattivi. Per ulteriori informazioni sulla creazione di un'immagine Bring Your Own License (BYOL) personalizzata, consulta la sezione «Risorse correlate».	Architetto F5, ingegnere, architetto cloud, ingegnere
Rafforza i dispositivi.	Ciò contribuirà a proteggere da potenziali vulnerabilità.	Architetto F5, ingegnere

Configura il tuo nuovo ambiente AWS

Operazione	Description	Competenze richieste
Crea account edge e di sicurezza.	Accedi a Console di gestione AWS e crea il servizio Account AWS che fornirà e gestirà i servizi edge e di sicurezza . Questi account potrebbero essere diversi dagli account utilizzati VPCs per servizi e applicazioni condivisi. Questo passaggio può essere	Architetto del cloud, ingegnere

Operazione	Description	Competenze richieste
	completato come parte di una landing zone.	
Implementa l'edge e la sicurezza VPCs.	Imposta e configura quanto VPCs necessario per fornire servizi perimetrali e di sicurezza.	Architetto del cloud, ingegnere
Connect al data center di origine.	Connect al data center di origine che ospita il carico di lavoro F5 BIG-IP.	Architetto del cloud, architetto di rete, ingegnere
Implementa le connessioni VPC.	Connect il servizio edge e di sicurezza VPCs all'appli cazione VPCs.	Architetto di rete, ingegnere
Distribuisci le istanze.	Distribuisci le istanze utilizzan do i CloudFormation modelli della sezione «Risorse correlate».	Architetto F5, ingegnere
Testa e configura il failover delle istanze.	Assicurati che il modello AWS Advanced HA iApp o l'estensione F5 Cloud Failover siano configurati e funzionino correttamente.	Architetto F5, ingegnere

Configurazione delle reti

Operazione	Description	Competenze richieste
Preparare la topologia VPC.	Apri la console Amazon VPC e assicurati che il tuo VPC disponga di tutte le sottoreti e le protezioni necessarie per	Architetto di rete, architetto F5, architetto cloud, ingegnere

Operazione	Description	Competenze richieste
	l'implementazione di F5 BIG-IP VE.	
Prepara i tuoi endpoint VPC.	Prepara gli endpoint VPC per Amazon, Amazon EC2 S3 e AWS STS se un carico di lavoro F5 BIG-IP non ha accesso a un gateway NAT o a un indirizzo IP elastico su un'interfaccia TMM.	Architetto del cloud, ingegnere

Migrazione dei dati

Operazione	Description	Competenze richieste
Migrare la configurazione.	Migrare la configurazione F5 BIG-IP a F5 BIG-IP VE su. Cloud AWS	Architetto F5, ingegnere
Associa il secondario IPs.	Gli indirizzi IP dei server virtuali hanno una relazione con gli indirizzi IP secondari assegnati alle istanze. Assegna indirizzi IP secondari e assicurati che sia seleziona ta l'opzione «Consenti rimappa/riassegnazione».	Architetto F5, ingegnere

Configurazioni di test

Operazione	Description	Competenze richieste
Convalida le configurazioni del server virtuale.	Prova i server virtuali.	F5 Architect, App Teams

Finalizza le operazioni

Operazione	Description	Competenze richieste
Crea la strategia di backup.	I sistemi devono essere spenti per creare un'istant anea completa. Per ulteriori informazioni, vedere «Aggiornamento di una macchina virtuale F5 BIG-IP» nella sezione «Risorse correlate».	Architetto F5, architetto cloud, ingegnere
Crea il runbook di failover del cluster.	Assicurati che il processo di failover runbook sia completo.	Architetto F5, ingegnere
Configura e convalida la registrazione.	Configura F5 Telemetry Streaming per inviare i log alle destinazioni richieste.	Architetto F5, ingegnere

Completa il cutover

Operazione	Description	Competenze richieste
Passate alla nuova installaz ione.		Architetto F5, architetto del cloud, architetto di rete, ingegnere, AppTeams

Risorse correlate

Guida alla migrazione

- Migrazione da F5 BIG-IP a F5 BIG-IP VE su Cloud AWS

Risorse F5

- CloudFormation modelli nel repository F5 GitHub

- F5 dentro Marketplace AWS
- Panoramica di F5 BIG-IP VE
- Esempio Quickstart - BIG-IP Virtual Edition con WAF (LTM + ASM)
- Servizi applicativi F5 su AWS: una panoramica (video)
- Guida per l'utente di F5 Application Services 3 Extension
- Documentazione sul cloud F5
- Wiki REST di F5 iControl
- F5 Panoramica dei singoli file di configurazione (11.x - 15.x)
- Whitepaper F5
- Strumento di generazione di immagini F5 BIG-IP
- Aggiornamento di una macchina virtuale F5 BIG-IP VE
- Panoramica dell'opzione «platform-migrate» dell'archivio UCS

Esegui la migrazione di un'applicazione web Go locale su AWS Elastic Beanstalk utilizzando il metodo binario

Suhas Basavaraj e Shumaz Mukhtar Kazi, Amazon Web Services

Riepilogo

Questo modello descrive come migrare un'applicazione web Go locale su AWS Elastic Beanstalk. Dopo la migrazione dell'applicazione, Elastic Beanstalk crea il file binario per il bundle di origine e lo distribuisce su un'istanza Amazon Elastic Compute Cloud (Amazon). EC2

Trattandosi di una strategia di migrazione rehost, l'approccio di questo pattern è rapido e non richiede modifiche al codice, il che significa meno tempo di test e migrazione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un'applicazione web Go locale.
- Un GitHub repository che contiene il codice sorgente dell'applicazione Go. Se non lo utilizzi GitHub, esistono altri modi per creare un bundle di sorgenti dell'applicazione per Elastic Beanstalk.

Versioni del prodotto

- La versione Go più recente supportata da Elastic Beanstalk. Per ulteriori informazioni, consulta la documentazione di Elastic Beanstalk.

Architecture

Stack tecnologico di origine

- Un'applicazione web Go locale

Stack tecnologico Target

- AWS Elastic Beanstalk
- Amazon CloudWatch

Architettura Target

Tools (Strumenti)

- AWS Elastic Beanstalk distribuisce e gestisce rapidamente le applicazioni nel cloud AWS senza che gli utenti debbano conoscere l'infrastruttura che esegue tali applicazioni. Elastic Beanstalk riduce la complessità della gestione senza limitare le scelte o il controllo.
- GitHubè un sistema di controllo delle versioni distribuito open source.

Epiche

Crea il file.zip del pacchetto sorgente dell'applicazione web Go

Operazione	Description	Competenze richieste
Crea il pacchetto sorgente per l'applicazione Go.	Apri il GitHub repository che contiene il codice sorgente dell'applicazione Go e prepara il pacchetto sorgente. Il pacchetto sorgente contiene un file `application.go` sorgente nella directory principale, che ospita il pacchetto principale dell'appl icazione Go. Se non lo utilizzat e GitHub, consultate la precedente sezione Prerequis iti di questo schema per scoprire altri modi per creare il bundle di sorgenti dell'appl icazione.	Amministratore di sistema, sviluppatore di applicazioni
Creazione di un file di configurazione.	Crea una `.ebextensions` cartella nel tuo pacchetto sorgente, quindi crea un `options.config` file all'interno di questa cartella.	Amministratore di sistema, sviluppatore di applicazioni

Operazione	Description	Competenze richieste
	Per ulteriori informazioni, consulta la documentazione di Elastic Beanstalk.	
Crea il file.zip del pacchetto sorgente.	Eseguire il seguente comando seguente. `git archive -o ../godemoapp.zip HEAD` Questo crea il file.zip del bundle di origine. Scaricate e salvate il file.zip come file locale. **Important** Il file.zip non può superare i 512 MB e non può includere una cartella principale o una directory di primo livello.	Amministratore di sistema, sviluppatore di applicazioni

Esegui la migrazione dell'applicazione web Go su Elastic Beanstalk

Operazione	Description	Competenze richieste
Scegli l'applicazione Elastic Beanstalk.	1. Accedi alla Console di gestione AWS e apri la console Elastic Beanstalk. 2. Dall'elenco delle regioni, scegli la tua regione AWS.	Amministratore di sistema, sviluppatore di applicazioni

Operazione	Description	Competenze richieste
	3. Nel riquadro di navigazione, scegli Applicazioni, quindi scegli un'applicazione Elastic Beanstalk esistente o creane una. Per istruzioni su come creare un'applicazione Elastic Beanstalk, consulta la documentazione di Elastic Beanstalk.	
Avvia l'ambiente del server web Elastic Beanstalk.	1. Nella pagina di panoramica dell'applicazione, scegli Crea un nuovo ambiente, quindi scegli Ambiente server Web. 2. Completa i campi Nome ambiente e Nome dominio. 3. Scegli la versione della piattaforma e seleziona Vai come piattaforma.	Amministratore di sistema, sviluppatore di applicazioni
Carica il file.zip del bundle sorgente su Elastic Beanstalk.	1. In Codice dell'applicazione, scegli Carica il codice, quindi scegli File locale. 2. Scegliete il file.zip che contiene il pacchetto sorgente. 3. Nell'etichetta della versione, assegna al file un nome univoco, quindi scegli Crea ambiente.	Amministratore di sistema, sviluppatore di applicazioni

Operazione	Description	Competenze richieste
Prova l'applicazione web Go distribuita.	Verrai reindirizzato alla pagina di panoramica dell'appl icazione Elastic Beanstalk . Nella parte superiore della panoramica, accanto a Environment ID, scegli l'URL che termina con `elasticbeanstalk.com` per accedere all'applicazione. L'applicazione deve utilizzar e questo nome nel file di configurazione come variabile di ambiente e visualizzarlo sulla pagina Web.	Amministratore di sistema, sviluppatore di applicazioni

risoluzione dei problemi

Problema	Soluzione
Impossibile accedere all'applicazione tramite un Application Load Balancer.	Controlla il gruppo target che contiene l'applica zione Elastic Beanstalk. Se non è integro, accedi all'istanza di Elastic Beanstalk e `nginx.conf` controlla la configurazione del file per verificare che venga indirizzato all'URL corretto dello stato di integrità. Potrebbe essere necessario modificare l'URL del controllo dello stato del gruppo target.

Risorse correlate

- Versioni della piattaforma Go supportate da Elastic Beanstalk
- Utilizzo dei file di configurazione con Elastic Beanstalk
- Creazione di un'applicazione di esempio in Elastic Beanstalk

Esegui la migrazione di un server SFTP locale a Using AWS AWS Transfer for SFTP

Akash Kumar, Amazon Web Services

Riepilogo

Questo modello descrive come migrare una soluzione di trasferimento file locale che utilizza il Secure Shell (SSH) File Transfer Protocol (SFTP) verso il utilizzando il servizio. Cloud AWS AWS Transfer for SFTP Gli utenti generalmente si connettono a un server SFTP tramite il relativo nome di dominio o tramite IP fisso. Questo modello copre entrambi i casi.

AWS Transfer for SFTP è membro di AWS Transfer Family. È un servizio di trasferimento sicuro che è possibile utilizzare per trasferire file da e verso i servizi di AWS archiviazione tramite SFTP. Puoi utilizzarlo AWS Transfer for SFTP con Amazon Simple Storage Service (Amazon S3) o Amazon Elastic File System (Amazon EFS). Questo modello utilizza Amazon S3 per lo storage.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS.
- Un nome di dominio SFTP esistente o un IP SFTP fisso.

Limitazioni

- L'oggetto più grande che è possibile trasferire in una richiesta è attualmente di 5 GiB. Per file di dimensioni superiori a 100 MiB, prendi in considerazione l'utilizzo del caricamento multiparte di Amazon S3.

Architecture

Stack tecnologico di origine

- File flat locali o file di dump del database.

Stack tecnologico Target

- AWS Transfer for SFTP
- Simple Storage Service (Amazon S3)

- Amazon Virtual Private Cloud (Amazon VPC) (Amazon VPC)
- AWS Identity and Access Management Ruoli e politiche (IAM)
- Indirizzi IP elastici
- Gruppi di sicurezza
- Amazon CloudWatch Logs (opzionale)

Architettura Target

Automazione e scalabilità

Per automatizzare l'architettura di destinazione per questo modello, utilizza i CloudFormation modelli allegati:

- `amazon-vpc-subnets.yml`fornisce un cloud privato virtuale (VPC) con due sottoreti pubbliche e due private.
- `amazon-sftp-server.yml`fornisce il server SFTP.
- `amazon-sftp-customer.yml`aggiunge utenti.

Tools (Strumenti)

Servizi AWS

- Amazon CloudWatch Logs ti aiuta a centralizzare i log di tutti i tuoi sistemi e applicazioni, Servizi AWS così puoi monitorarli e archiviarli in modo sicuro.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati. Questo modello utilizza Amazon S3 come sistema di storage per i trasferimenti di file.
- AWS Transfer for SFTPti aiuta a trasferire file da e verso i servizi di AWS storage tramite il protocollo SFTP.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.

Epiche

Crea un VPC

Operazione	Description	Competenze richieste
Crea un VPC con sottoreti.	Apri la Console Amazon VPC. Crea un cloud privato virtuale (VPC) con due sottoreti pubbliche. (La seconda sottorete offre un'elevata disponibilità.) oppure Puoi implementare il CloudFormation modello allegato nella CloudFormation console per automatizzare le attività di questa epopea. `amazon-vpc-subnets.yml`	Sviluppatore, amministratore di sistema
Aggiungi un gateway Internet.	Esegui il provisioning di un gateway Internet e collegalo al VPC.	Sviluppatore, amministratore di sistema
Esegui la migrazione di un IP esistente.	Collega un IP esistente all'indirizzo IP elastico. Puoi creare un indirizzo IP elastico dal tuo pool di indirizzi e utilizzarlo.	Sviluppatore, amministratore di sistema

Esegui il provisioning di un server SFTP

Operazione	Description	Competenze richieste
Crea un server SFTP.	Apri la AWS Transfer Family console. Segui le istruzioni	Sviluppatore, amministratore di sistema

Operazione	Description	Competenze richieste
	riportate nella AWS Transfer Family documentazione relativa alla creazione di un endpoint con accesso a Internet per il tuo server per creare un server SFTP con un endpoint connesso a Internet. Per il tipo di endpoint, scegli VPC ospitato. Per Access, scegli Internet Facing. Per VPC, scegli il VPC che hai creato nell'epopea precedente. oppure Puoi implementare il CloudFormation modello allegato nella CloudFormation console per automatizzare le attività di questa epopea. `amazon-sftp-server.yml`	
Esegui la migrazione del nome di dominio.	Allega il nome di dominio esistente al nome host personalizzato. Se utilizzi un nuovo nome di dominio, usa l'alias DNS di Amazon Route 53. Per un nome di dominio esistente, scegli Altro DNS. Per ulteriori informazioni, consulta Lavorare con nomi host personalizzati nella AWS Transfer Family documenta zione.	Sviluppatore, amministratore di sistema

Operazione	Description	Competenze richieste
Aggiungi un ruolo CloudWatch di registrazione.	(Facoltativo) se desideri abilitare CloudWatch la registrazione, crea un `Transfer` ruolo con le operazioni dell'API CloudWatch Logs`logs:CreateLogGroup` ,`logs:CreateLogStream` , `logs:DescribeLogStreams` e. `logs:PutLogEvents` Per ulteriori informazioni, consulta Log activity with CloudWatch nella AWS Transfer Family documentazione.	Sviluppatore, amministratore di sistema
Salva e invia.	Scegli Save (Salva). Per Azioni, scegli Avvia e attendi che il server SFTP venga creato con lo stato Online.	Sviluppatore, amministratore di sistema

Mappa gli indirizzi IP elastici sul server SFTP

Operazione	Description	Competenze richieste
Arresta il server in modo da poter modificare le impostazioni.	Sulla AWS Transfer Family console, scegli Server, quindi seleziona il server SFTP che hai creato. In Actions (Operazioni), scegliere Stop (Arresta). Quando il server è offline, scegli Modifica per modificarne le impostazioni.	Sviluppatore, amministratore di sistema

Operazione	Description	Competenze richieste
Scegli zone di disponibilità e sottoreti.	Nella sezione Zone di disponibilità, scegli le zone di disponibilità e le sottoreti per il tuo VPC.	Sviluppatore, amministratore di sistema
Aggiungi indirizzi IP elastici.	Per IPv4 Indirizzi, scegli un indirizzo IP elastico per ogni sottorete, quindi scegli Salva.	Sviluppatore, amministratore di sistema

Aggiungere utenti

Operazione	Description	Competenze richieste
Crea un ruolo IAM per consentire agli utenti di accedere al bucket S3.	Crea un ruolo IAM `Transfer` e aggiungi `s3:ListBucket` `s3:GetBucketLocation` , e `s3:PutObject` con il nome del bucket S3 come risorsa. Per ulteriori informazioni, consulta Creare un ruolo e una policy IAM nella AWS Transfer Family documentazione. oppure Puoi implementare il CloudFormation modello allegato nella CloudFormation console per automatizzare le attività di questa epopea. `amazon-sftp-customer.yml`	Sviluppatore, amministratore di sistema

Operazione	Description	Competenze richieste
Crea un bucket S3.	Crea un bucket S3 per l'applicazione.	Sviluppatore, amministratore di sistema
Crea cartelle opzionali.	(Facoltativo) Se desideri archiviare i file per gli utenti separatamente, in cartelle Amazon S3 specifiche, aggiungi le cartelle appropria te.	Sviluppatore, amministratore di sistema
Crea una chiave pubblica SSH.	Per creare una coppia di chiavi SSH, consulta Generare chiavi SSH nella AWS Transfer Family documenta zione.	Sviluppatore, amministratore di sistema
Aggiungere gli utenti.	Sulla AWS Transfer Family console, scegli Server, seleziona il server SFTP che hai creato, quindi scegli Aggiungi utente. Per Home directory, scegli il bucket S3 che hai creato. Per la chiave pubblica SSH, specificare la parte della chiave pubblica della coppia di chiavi SSH. Aggiungi utenti per il server SFTP, quindi scegli Aggiungi.	Sviluppatore, amministratore di sistema

Prova il server SFTP

Operazione	Description	Competenze richieste
Aggiorna il gruppo di sicurezza .	Nella sezione Gruppi di sicurezza del server SFTP,	Developer

Operazione	Description	Competenze richieste
	aggiungi l'IP della macchina di test per ottenere l'accesso SFTP.	
Utilizzate un'utilità client SFTP per testare il server.	Eseguite il test dei trasferimenti di file utilizzando qualsiasi utilità client SFTP. Per un elenco di client e istruzioni, consultate Trasferimento di file utilizzando un client nella AWS Transfer Family documentazione.	Developer

Risorse correlate

- AWS Transfer Family Guida per l'utente
- Guida per l'utente di Amazon S3
- Indirizzi IP elastici nella EC2 documentazione di Amazon

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Migra una macchina virtuale locale su Amazon EC2 utilizzando AWS Application Migration Service

Thanh Nguyen, Amazon Web Services

Riepilogo

Per quanto riguarda la migrazione delle applicazioni, le organizzazioni possono adottare approcci diversi per riospitare (lift and shift) i server dell'applicazione dall'ambiente locale al cloud Amazon Web Services (AWS). Un modo consiste nel fornire nuove istanze Amazon Elastic Compute Cloud (Amazon EC2) e quindi installare e configurare l'applicazione da zero. Un altro approccio consiste nell'utilizzare servizi di migrazione nativi di AWS o di terze parti per migrare più server contemporaneamente.

Questo modello descrive i passaggi per la migrazione di una macchina virtuale (VM) supportata su un' EC2 istanza Amazon sul cloud AWS utilizzando AWS Application Migration Service. Puoi utilizzare l'approccio descritto in questo modello per migrare una o più macchine virtuali manualmente, una per una o automaticamente creando script di automazione appropriati in base ai passaggi descritti.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo in una delle regioni AWS che supportano Application Migration Service
- Connettività di rete tra il server di origine e il EC2 server di destinazione tramite una rete privata utilizzando AWS Direct Connect o una rete privata virtuale (VPN) o tramite Internet

Limitazioni

- Per l'elenco aggiornato delle regioni supportate, consulta le regioni AWS supportate.
- Per un elenco dei sistemi operativi supportati, consulta la sezione Sistemi operativi supportati e la sezione Generale di Amazon EC2 FAQs.

Architecture

Stack tecnologico di origine

- Un server fisico, virtuale o ospitato nel cloud che esegue un sistema operativo supportato da Amazon EC2

Stack tecnologico Target

- Un' EC2 istanza Amazon che esegue lo stesso sistema operativo della macchina virtuale di origine
- Amazon Elastic Block Store (Amazon EBS)

Architettura di origine e destinazione

Il diagramma seguente mostra l'architettura di alto livello e i componenti principali della soluzione. Nel data center locale sono presenti macchine virtuali con dischi locali. In AWS, è disponibile un'area di staging con server di replica e un'area di risorse migrate con EC2 istanze per test e cutover. Entrambe le sottoreti contengono volumi EBS.

1. Inizializza AWS Application Migration Service.
2. Configura la configurazione e il reporting del server dell'area di staging, incluse le risorse dell'area di staging.
3. Installa gli agenti sui server di origine e utilizza la replica continua dei dati a livello di blocco (compressa e crittografata).
4. Automatizza l'orchestrazione e la conversione del sistema per abbreviare la finestra intermedia.

Architettura di rete

Il diagramma seguente mostra l'architettura di alto livello e i componenti principali della soluzione dal punto di vista della rete, inclusi i protocolli e le porte necessari per la comunicazione tra i componenti principali nel data center locale e su AWS.

Tools (Strumenti)

- AWS Application Migration Service ti aiuta a riospitare (lift and shift) le applicazioni nel cloud AWS senza modifiche e con tempi di inattività minimi.

Best practice

- Non mettere offline il server di origine né eseguire un riavvio fino al completamento del cutover sull'istanza di destinazione EC2 .
- Offri agli utenti ampie opportunità di eseguire test di accettazione degli utenti (UAT) sul server di destinazione per identificare e risolvere eventuali problemi. Idealmente, questo test dovrebbe iniziare almeno due settimane prima del termine.
- Monitora frequentemente lo stato della replica del server sulla console di Application Migration Service per identificare tempestivamente i problemi.
- Utilizza credenziali AWS Identity and Access Management (IAM) temporanee per l'installazione dell'agente anziché credenziali utente IAM permanenti.

Epiche

Generazione di credenziali AWS

Operazione	Description	Competenze richieste
Crea il ruolo IAM di AWS Replication Agent.	Accedi con autorizzazioni amministrative all'account AWS. Sulla console AWS Identity and Access Management (IAM), crea un ruolo IAM: 1. Sulla console IAM, scegli Roles. 2. Scegli Crea ruolo. 3. Nella pagina Seleziona entità affidabile, nella sezione Tipo di entità affidabile, seleziona Account AWS. 4. Nella sezione Un account AWS, seleziona Questo account (< account-id>).	Amministratore AWS, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	5. Scegli Next (Successivo). 6. Nella pagina Aggiungi autorizzazioni, cerca la `AWSApplicationMigrationAgentInstallationPolicy` policy, seleziona la casella di controllo accanto al nome della policy. 7. Scegli Next (Successivo). 8. Nella pagina dei dettagli del ruolo, immettete MGN_Agent_Installation_Role come nome del ruolo. 9. Verifica che i campi siano corretti, quindi scegli Crea ruolo.	

Operazione	Description	Competenze richieste
Genera credenziali di sicurezza temporanee.	Su una macchina con AWS Command Line Interface (AWS CLI) installata, accedi con autorizzazioni amministrative. In alternativa (all'interno di una regione AWS supportata), sulla Console di gestione AWS, accedi con autorizzazioni amministrative all'account AWS e apri AWS CloudShell. Genera credenziali temporanee con il seguente comando, sostituendolo `<account-id>` con l'ID dell'account AWS. `aws sts assume-role --role-arn arn:aws:iam::<account-id>:role/MGN_Agent_Installation_Role --role-session-name mgn_installation_session_role` Dall'output del comando, copia i valori per `AccessKeyId` **`SecretAccessKey`** , e **`SessionToken`** . Conservali in un luogo sicuro per un uso successivo.	Amministratore AWS, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	Important Queste credenzia li temporanee scadranno dopo un'ora. Se hai bisogno di credenzia li dopo un'ora, ripeti i passaggi precedenti.	

Inizializza Application Migration Service e crea il modello di impostazioni di replica

Operazione	Description	Competenze richieste
Inizializza il servizio.	Sulla console, accedi con le autorizzazioni amministrative all'account AWS. Scegli Application Migration Service, quindi scegli Inizia.	Amministratore AWS, ingegnere addetto alla migrazione
Crea e configura il modello di impostazioni di replica.	1. Fornisci i seguenti dettagli di configurazione: a. Seleziona la sottorete dell'area di staging. b. Seleziona il tipo di istanza del server di replica (`t3.small`per impostazione predefini ta). c. Seleziona il tipo di volume EBS (gp3 per impostazione predefini ta).	Amministratore AWS, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	d. Seleziona l'opzione di crittografia EBS. e. Assicurati che la casella di controllo Usa sempre il gruppo di sicurezza Application Migration Service sia selezionata. f. Seleziona la casella di controllo Usa IP privato per la replica dei dati (VPN DirectConnect, peering VPC) se utilizzi la connettività di rete privata tra l'ambiente locale e AWS. g. Seleziona la casella di controllo Throttle network width (per server, in Mbps) se desideri limitare la larghezza di banda di rete per Application Migration Service. 2. Scegli Crea modello. Application Migration Service creerà automaticamente tutti i ruoli IAM necessari per facilitare la replica dei dati e l'avvio dei server migrati.	

Installa gli agenti di replica AWS sui computer di origine

Operazione	Description	Competenze richieste
Tieni a portata di mano le credenziali AWS richieste.	Quando esegui il file di installazione su un server di origine, dovrai inserire le credenziali temporane e generate in precedenz a, tra cui`AccessKeyId` , `SecretAccessKey` e. `SessionToken`	Ingegnere addetto alla migrazione, amministratore AWS
Per i server Linux, installa l'agente.	Copia il comando di installaz ione, accedi ai server di origine ed esegui il programma di installazione. Per istruzioni dettagliate, consulta la documentazione AWS.	Amministratore AWS, ingegnere addetto alla migrazione
Per i server Windows, installa l'agente.	Scarica il file di installazione su ogni server, quindi esegui il comando di installazione. Per istruzioni dettagliate, consulta la documentazione AWS.	Amministratore AWS, ingegnere addetto alla migrazione
Attendi il completamento della replica iniziale dei dati.	Una volta installato l'agente, il server di origine verrà visualizzato nella console di Application Migration Service, nella sezione Server di origine. Attendi che il server venga sottoposto alla replica iniziale dei dati.	Amministratore AWS, ingegnere addetto alla migrazione

Configura le impostazioni di avvio

Operazione	Description	Competenze richieste
Specificare i dettagli del server.	Nella console di Application Migration Service, scegli la sezione Server di origine, quindi scegli un nome di server dall'elenco per accedere ai dettagli del server.	Amministratore AWS, ingegnere addetto alla migrazione
Configura le impostazioni di avvio.	Scegli la scheda Impostazioni di avvio. Puoi configurare una varietà di impostazioni, incluse le impostazioni generali di avvio e le impostazioni del modello di EC2 avvio. Per istruzioni dettagliate, consulta la documentazione AWS.	Amministratore AWS, ingegnere addetto alla migrazione

Esegui un test

Operazione	Description	Competenze richieste
Prova i server di origine.	1. Nella console Application Migration Service, nella sezione Server di origine, assicurati che il ciclo di vita della migrazione dei server di origine sia pronto per il test e che lo stato della replica dei dati sia integro. 2. Seleziona la casella di controllo a sinistra di ogni server di origine.	Amministratore AWS, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	3. Scegli Test e Cutover, quindi scegli Launch Test Instance. 4. Quando richiesto, scegli Launch. I server verranno avviati.	
Verifica che il test sia stato completato con successo.	Una volta avviato completamente il server di test, lo stato Avvisi nella pagina mostrerà Avviato per ogni server.	Amministratore AWS, ingegnere addetto alla migrazione
Testa il server.	Esegui test sul server di test per assicurarti che funzioni come previsto.	Amministratore AWS, ingegnere addetto alla migrazione

Pianifica ed esegui un cutover

Operazione	Description	Competenze richieste
Pianifica una finestra di taglio.	Pianifica un periodo di tempo limite adeguato con i team competenti.	Amministratore AWS, ingegnere addetto alla migrazione
Esegui il cutover.	1. Nella console di migrazione delle applicazioni, nella pagina Server di origine, seleziona la casella di controllo a sinistra di ogni server di origine. 2. Scegli Test e Cutover e seleziona Contrassegna come «Pronto per il taglio».	Amministratore AWS, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	3. Verifica che il ciclo di vita della migrazione di ogni server di origine sia pronto per il cutover. 4. Scegli Test e Cutover, quindi seleziona Launch cutover instances. 5. Quando richiesto, scegli Launch. I server verranno avviati. Il ciclo di vita della migrazione del server di origine passerà a Cutover in corso.	
Verifica che il cutover sia stato completato correttamente.	Dopo l'avvio completo dei server di cutover, lo stato Avvisi nella pagina Server di origine mostrerà Avviato per ogni server.	Amministratore AWS, ingegnere addetto alla migrazione
Testa il server.	Esegui dei test sul server cutover per assicurarti che funzioni come previsto.	Amministratore AWS, ingegnere addetto alla migrazione
Finalizza il cutover.	Scegli Test e Cutover, quindi seleziona Finalize cutover per finalizzare il processo di migrazione.	Amministratore AWS, ingegnere addetto alla migrazione

Risorse correlate

- AWS Servizio della migrazione di applicazioni
- Guida per l'utente di AWS Application Migration Service

Esegui la migrazione di piccoli set di dati da locale ad Amazon S3 utilizzando AWS SFTP

Charles Gibson e Sergiy Shevchenko, Amazon Web Services

Riepilogo

Questo modello descrive come migrare piccoli set di dati (5 TB o meno) dai data center locali ad Amazon Simple Storage Service (Amazon S3) utilizzando (). AWS Transfer for SFTP AWS SFTP I dati possono essere dump di database o file flat.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un AWS Direct Connect collegamento stabilito tra il data center e AWS

Limitazioni

- I file di dati devono pesare meno di 5 TB. Per file superiori a 5 TB, puoi eseguire un caricamento in più parti su Amazon S3 o scegliere un altro metodo di trasferimento dei dati.

Architecture

Stack tecnologico di origine

- File flat o dump di database locali

Stack tecnologico Target

- Simple Storage Service (Amazon S3)

Architettura di origine e destinazione

Tools (Strumenti)

- AWS SFTP— Consente il trasferimento di file direttamente da e verso Amazon S3 utilizzando Secure File Transfer Protocol (SFTP).

- AWS Direct Connect— Stabilisce una connessione di rete dedicata dai data center locali a. AWS
- Endpoint VPC: consentono di connettere privatamente un VPC a servizi endpoint VPC supportati Servizi AWS e forniti AWS PrivateLink da senza un gateway Internet, un dispositivo NAT (Network Address Translation), una connessione VPN o una connessione. Direct Connect Le istanze in un VPC non richiedono indirizzi IP pubblici per comunicare con le risorse del servizio.

Epiche

Preparati per la migrazione

Operazione	Description	Competenze richieste
Documenta gli attuali requisiti SFTP.		Proprietario dell'applicazione, SA
Identifica i requisiti di autentica zione.	I requisiti possono includere l'autenticazione basata su chiave, il nome utente o la password o il provider di identità (IdP).	Proprietario dell'applicazione, SA
Identifica i requisiti di integrazi one delle applicazioni.		Proprietario dell'applicazione
Identifica gli utenti che richiedono il servizio.		Proprietario dell'applicazione
Determina il nome DNS per l'endpoint del server SFTP.		Rete
Determinare la strategia di backup.		SA, DBA (se i dati vengono trasferiti)
Identifica la migrazione delle applicazioni o la strategia di cutover.		Proprietario dell'applicazione, SA, DBA

Configura l'infrastruttura

Operazione	Description	Competenze richieste
Crea uno o più cloud privati virtuali (VPCs) e sottoreti nel tuo. Account AWS		Proprietario dell'applicazione, AMS
Crea i gruppi di sicurezza e l'elenco di controllo degli accessi alla rete (ACL).		Sicurezza, rete, AMS
Crea il bucket Amazon S3.		Proprietario dell'applicazione, AMS
Crea il ruolo AWS Identity and Access Management (IAM).	Crea una policy IAM che includa le autorizzazioni per consentire l'accesso AWS SFTP al tuo bucket Amazon S3. Questa policy IAM determina il livello di accesso da fornire agli utenti SFTP. Crea un'altra policy IAM con AWS SFTP cui stabilire una relazione di fiducia.	Sicurezza, AMS
Associa un dominio registrato (opzionale).	Se hai il tuo dominio registrat o, puoi associarlo al server SFTP. È possibile indirizzare il traffico SFTP verso l'endpoin t del server SFTP da un dominio o da un sottodominio.	Rete, AMS
Crea un server SFTP.	Specificate il tipo di provider di identità utilizzato dal servizio per autenticare gli utenti.	Proprietario dell'applicazione, AMS

Operazione	Description	Competenze richieste
Aprire un client SFTP.	Aprire un client SFTP e configurare la connessione per utilizzare l'host dell'endp oint SFTP. AWS SFTP supporta qualsiasi client SFTP standard. I client SFTP più utilizzati includono OpenSSH, WinSCP, Cyberduck e. FileZilla È possibile ottenere il nome host del server SFTP dalla console. AWS SFTP	Proprietario dell'applicazione, AMS

Pianifica e testa

Operazione	Description	Competenze richieste
Pianifica la migrazione delle applicazioni.	Pianifica le modifiche necessarie alla configura zione dell'applicazione, imposta la data di migrazione e determina la pianificazione del test.	Proprietario dell'applicazione, AMS
Testa l'infrastruttura.	Esegui il test in un ambiente non di produzione.	Proprietario dell'applicazione, AMS

Risorse correlate

Riferimenti

- AWS Transfer for SFTP Guida per l'utente
- AWS Direct Connect resources
- Endpoint VPC

Tutorial e video

- AWS Transfer for SFTP (video)
- AWS Transfer for SFTP guida per l'utente
- AWS SA Whiteboarding - Direct Connect (video)

Esegui la migrazione di un database Oracle locale a Oracle su Amazon EC2

Baji Shaik e Pankaj Choudhary, Amazon Web Services

Riepilogo

Questo modello illustra i passaggi per la migrazione di un database Oracle locale a Oracle su un'istanza Amazon Elastic Compute Cloud (Amazon EC2). Descrive due opzioni per la migrazione: utilizzando AWS Data Migration Service (AWS DMS) o utilizzando strumenti Oracle nativi come RMAN, Data Pump import/export, tablespace trasportabili e Oracle. GoldenGate

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database Oracle di origine in un data center locale

Limitazioni

- Il sistema operativo (OS) di destinazione deve essere supportato da Amazon EC2. Per un elenco completo dei sistemi supportati, consulta Amazon EC2 FAQs.

Versioni del prodotto

- Oracle Database versioni 10.2 e successive (per le versioni 10.x), 11g e fino a 12.2 e 18c per le edizioni Enterprise, Standard, Standard One e Standard Two Per l'elenco più recente delle versioni supportate da AWS DMS, consulta «Database locali e di EC2 istanze Amazon» in Sources for Data Migration nella documentazione di AWS DMS.

Architecture

Stack tecnologico di origine

- Un database Oracle locale

Stack tecnologico Target

- Un'istanza di database Oracle su Amazon EC2

Architettura Target

Architettura di migrazione dei dati

Utilizzando AWS DMS:

Utilizzo di strumenti Oracle nativi:

Tools (Strumenti)

- AWS DMS - AWS Database Migration Services (AWS DMS) supporta diversi tipi di database di origine e destinazione. Per informazioni sulle versioni e le edizioni del database supportate, consulta Using an Oracle Database as a Source for AWS DMS. Ti consigliamo di utilizzare la versione più recente di AWS DMS per il supporto più completo della versione e delle funzionalità.
- Strumenti Oracle nativi: RMAN, importazione/esportazione di Data Pump, tablespace trasportabili, Oracle GoldenGate

Epiche

Pianifica la migrazione

Operazione	Description	Competenze richieste
Convalida le versioni dei database di origine e di destinazione.		DBA
Identifica la versione del sistema operativo di destinazione.		DBA, SysAdmin
Identifica i requisiti hardware per l'istanza del server di destinazione in base all'elenco di compatibilità e ai requisiti di capacità di Oracle.		DBA, SysAdmin

Operazione	Description	Competenze richieste
Identifica i requisiti di archiviazione (tipo e capacità di archiviazione).		DBA, SysAdmin
Identifica i requisiti di rete (latenza e larghezza di banda).		DBA, SysAdmin
Scegli il tipo di istanza corretto in base alla capacità, alle funzionalità di archiviazione e alle funzionalità di rete.		DBA, SysAdmin
Identifica i requisiti di sicurezza degli network/host accessi per i database di origine e di destinazione.		DBA, SysAdmin
Identifica un elenco di utenti del sistema operativo necessari per l'installazione del software Oracle.		DBA, SysAdmin
Scarica AWS Schema Conversion Tool (AWS SCT) e i driver.		DBA
Crea un progetto AWS SCT per il carico di lavoro e connettiti al database di origine.		DBA
Genera file SQL per la creazione di oggetti (tabelle, indici, sequenze, ecc.).		DBA

Operazione	Description	Competenze richieste
Determinare una strategia di backup.		DBA, SysAdmin
Determinare i requisiti di disponibilità.		DBA
Identifica la strategia di migrazione/commutazione delle applicazioni.		DBA, proprietario dell'app SysAdmin

Configura l'infrastruttura

Operazione	Description	Competenze richieste
Crea un cloud privato virtuale (VPC) e sottoreti nel tuo account AWS.		SysAdmin
Crea gruppi di sicurezza e liste di controllo degli accessi alla rete (). ACLs		SysAdmin
Configura e avvia l' EC2 istanza.		SysAdmin

Installa il software Oracle

Operazione	Description	Competenze richieste
Crea gli utenti e i gruppi del sistema operativo necessari per il software Oracle.		DBA, SysAdmin
Scarica la versione richiesta del software Oracle.		

Operazione	Description	Competenze richieste
Installa il software Oracle sull' EC2 istanza.		DBA, SysAdmin
Crea oggetti come tabelle, chiavi primarie, viste e sequenze utilizzando gli script generati da AWS SCT.		DBA

Migrazione dei dati - opzione 1

Operazione	Description	Competenze richieste
Utilizza strumenti Oracle nativi o strumenti di terze parti per migrare oggetti e dati del database.	Gli strumenti Oracle includono l'importazione/esportazione di Data Pump, RMAN, tablespac e trasportabili e. GoldenGate	DBA

Migrazione dei dati - opzione 2

Operazione	Description	Competenze richieste
Determinare il metodo di migrazione.		DBA
Crea un'istanza di replica nella console AWS DMS.		DBA
Crea endpoint di origine e destinazione.		DBA
Creare un'attività di replica.		DBA
Abilita Change Data Capture (CDC) per acquisire le		DBA

Operazione	Description	Competenze richieste
modifiche per una replica continua.		
Esegui l'attività di replica e monitora i log.		DBA
Crea oggetti secondari come indici e chiavi esterne al termine del caricamento completo.		DBA

Migrare l'applicazione

Operazione	Description	Competenze richieste
Segui la strategia di migrazion e delle applicazioni.		DBA SysAdmin, proprietario dell'app

Tagliare

Operazione	Description	Competenze richieste
Segui la strategia di cutover/s witch-over dell'applicazione.		DBA, proprietario dell'app SysAdmin

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse temporanee di AWS Secrets Manager.		DBA, SysAdmin
Rivedi e convalida i documenti del progetto.		DBA, proprietario dell' SysAdminapp

Operazione	Description	Competenze richieste
Raccogli le metriche in tempo utile per la migrazione, percentuale di utilizzo manuale rispetto allo strumento, risparmi sui costi, ecc.		DBA, proprietario dell'app SysAdmin
Chiudi il progetto e fornisci feedback.		

Risorse correlate

Riferimenti

- Strategie per la migrazione dei database Oracle su AWS
- Migrazione dei database Oracle sul cloud AWS
- EC2 Sito web Amazon
- Sito web AWS DMS
- Post sul blog di AWS DMS
- EC2 Prezzi Amazon
- Licenza del software Oracle nell'ambiente di cloud computing

Tutorial e video

- Guida introduttiva ad Amazon EC2
- Guida introduttiva ad AWS DMS
- Introduzione ad Amazon EC2 - Elastic Cloud Server e hosting con AWS (video)

Esegui la migrazione di un database Oracle locale su Amazon EC2 utilizzando Oracle Data Pump

Navakanth Talluri, Amazon Web Services

Riepilogo

Durante la migrazione dei database, è necessario considerare fattori quali i motori e le versioni dei database di origine e di destinazione, gli strumenti e i servizi di migrazione e i periodi di inattività accettabili. Se stai migrando un database Oracle locale su Amazon Elastic Compute Cloud (Amazon EC2), puoi utilizzare strumenti Oracle, come Oracle Data Pump e Oracle Recovery Manager (RMAN). Per ulteriori informazioni sulle strategie, consulta Migrazione dei database Oracle al cloud AWS.

Oracle Data Pump ti aiuta a estrarre il backup logico e coerente del database e a ripristinarlo sull' EC2 istanza di destinazione. Questo modello descrive come migrare un database Oracle locale su un' EC2 istanza utilizzando Oracle Data Pump e il `NETWORK_LINK` parametro, con tempi di inattività minimi. Il `NETWORK_LINK` parametro avvia un'importazione tramite un collegamento al database. Il client Oracle Data Pump Import (impdp) sull' EC2 istanza di destinazione si connette al database di origine, recupera i dati da esso e li scrive direttamente nel database sull'istanza di destinazione. In questa soluzione non vengono utilizzati file di backup o dump.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un database Oracle locale che:
  - Non è un database Oracle Real Application Clusters (RAC)
  - Non è un database Oracle Automatic Storage Management (Oracle ASM)
  - È in modalità lettura-scrittura.
- Hai creato un collegamento AWS Direct Connect tra il tuo data center locale e AWS. Per ulteriori informazioni, consulta Creare una connessione (documentazione Direct Connect).

Versioni del prodotto

- Oracle Database 10g release 1 (10.1) e successive

Architecture

Stack tecnologico di origine

- Un server di database Oracle autonomo (non RAC e non ASM) in un data center locale

Stack tecnologico Target

- Un database Oracle in esecuzione su Amazon EC2

Architettura Target

Il pilastro dell'affidabilità di AWS Well-Architected Framework consiglia di creare backup dei dati per contribuire a fornire disponibilità e resilienza elevate. Per ulteriori informazioni, consulta Architecting for high availability in Best Practices for Running Oracle Database on AWS. Questo modello imposta i database primari e in standby sulle EC2 istanze utilizzando Oracle Active Data Guard. Per un'elevata disponibilità, le EC2 istanze devono trovarsi in zone di disponibilità diverse. Tuttavia, le zone di disponibilità possono trovarsi nella stessa regione AWS o in diverse regioni AWS.

Active Data Guard fornisce l'accesso in sola lettura a un database fisico in standby e applica continuamente le modifiche da ripetere dal database principale. In base al Recovery Point Objective (RPO) e al Recovery Time Objective (RTO), puoi scegliere tra opzioni di redo transport sincrone e asincrono.

L'immagine seguente mostra l'architettura di destinazione se le EC2 istanze primarie e di standby si trovano in diverse regioni AWS.

Architettura di migrazione dei dati

Dopo aver completato la configurazione dell'architettura di destinazione, si utilizza Oracle Data Pump per migrare i dati e gli schemi locali sull'istanza principale. EC2 Durante il cutover, le applicazioni non possono accedere al database locale o al database di destinazione. Queste applicazioni vengono chiuse finché non possono essere connesse al nuovo database di destinazione sull'istanza principale EC2 .

L'immagine seguente mostra l'architettura durante la migrazione dei dati. In questa architettura di esempio, le EC2 istanze primarie e di standby si trovano in diverse regioni AWS.

Tools (Strumenti)

Servizi AWS

- AWS Direct Connect collega la rete interna a una posizione Direct Connect tramite un cavo Ethernet standard in fibra ottica. Con questa connessione, puoi creare interfacce virtuali direttamente ai servizi AWS pubblici bypassando i provider di servizi Internet nel tuo percorso di rete.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.

Altri strumenti e servizi

- Oracle Active Data Guard ti aiuta a creare, mantenere, gestire e monitorare i database in standby.
- Oracle Data Pump ti aiuta a spostare dati e metadati da un database all'altro a velocità elevate.

Best practice

- Best Practice per l'esecuzione di Oracle Database su AWS
- Importazione di dati utilizzando NETWORK_LINK

Epiche

Configura le EC2 istanze su AWS

Operazione	Description	Competenze richieste
Identifica la configurazione hardware di origine per l'host locale e i parametri del kernel.	Convalida la configurazione locale, comprese le dimension i dello storage, input/output le operazioni al secondo (IOPS) e la CPU. Questo è important e per le licenze Oracle, che si basano sui core della CPU.	DBA, SysAdmin
Crea l'infrastruttura su AWS.	Crea cloud privati virtuali (VPCs), sottoreti private,	Amministratore di sistema DBA, AWS

Operazione	Description	Competenze richieste
	gruppi di sicurezza, elenchi di controllo degli accessi alla rete (ACLs), tabelle di routing e gateway Internet. Per ulteriori informazioni, consulta gli argomenti seguenti: • VPCs e sottoreti • Tutorial: creare un VPC da utilizzare con un'istanza di database	

Operazione	Description	Competenze richieste
Configura le EC2 istanze utilizzando Active Data Guard.	Configura EC2 le istanze AWS utilizzando una configura zione Active Data Guard, come descritto in AWS Well-Architected Framework. La versione di Oracle Database sull' EC2 istanza può essere diversa dalla versione locale perché questo modello utilizza backup logici. Tenere presente quanto segue: • Metti il database di destinazi one in modalità lettura-s crittura. • Nel database di destinazi one, fornisci i dettagli del Transparent Network Substrate (TNS) per il database di origine. Per ulteriori informazioni, consulta: • Avvio di un database (documentazione Oracle) • Creazione e configura zione di un database Oracle (documentazione Oracle)	Amministratore di sistema DBA, AWS

Esegui la migrazione del database su Amazon EC2

Operazione	Description	Competenze richieste
Crea un dblink al database locale dall'istanza. EC2	Crea un database link (dblink) tra il database Oracle sull' EC2 istanza e il database Oracle locale. Per ulteriori informazioni, vedere Utilizzo dell'importazione da Network Link per spostare i dati (documentazione Oracle).	DBA
Verifica la connessione tra l' EC2 istanza e l'host locale.	Usa il dblink per confermare che la connessione tra l' EC2 istanza e il database locale funzioni. Per istruzioni, consulta CREATE DATABASE LINK (documentazione Oracle).	DBA
Arresta tutte le applicazioni connesse al database locale.	Dopo l'approvazione del periodo di inattività del database, chiudi tutte le applicazioni e i job dipendenti che si connettono al database locale. Puoi farlo direttamente dall'applicazione o dal database usando cron. Per ulteriori informazioni, consulta Utilizzare l'utilità Crontab per pianificare le attività su Oracle Linux.	DBA, sviluppatore di app
Pianifica il processo di migrazione dei dati.	Sull'host di destinazione, usa il comando `impdb` per pianificare l'importazione di Data Pump. Questo collega	DBA

Operazione	Description	Competenze richieste
	il database di destinazione all'host locale e avvia la migrazione dei dati. Per ulteriori informazioni, vedere Data Pump Import e NETWORK_LINK (documentazione Oracle).	
Convalida la migrazione dei dati.	La convalida dei dati è un passaggio fondamentale. Per la convalida dei dati, puoi utilizzare strumenti personalizzati o strumenti Oracle, come una combinazione di query dblink e SQL.	DBA

Tagliare

Operazione	Description	Competenze richieste
Metti il database di origine in modalità di sola lettura.	Verificate che l'applicazione sia chiusa e che non siano state apportate modifiche al database di origine. Aprire il database di origine in modalità di sola lettura. In questo modo è possibile evitare transazioni aperte. Per ulteriori informazioni, vedere `ALTER DATABASE` nelle istruzioni SQL (documentazione Oracle).	DBA, DevOps ingegnere, sviluppatore di app
Convalida il conteggio e i dati degli oggetti.	Per convalidare i dati e l'oggetto, utilizza strumenti personalizzati o strumenti	DBA, sviluppatore di app

Operazione	Description	Competenze richieste
	Oracle, come una combinazi one di query dblink e SQL.	
Connect le applicazioni al database sull' EC2 istanza principale.	Modificate l'attributo di connessione dell'applicazione in modo che punti al nuovo database creato sull' EC2 istanza principale.	DBA, sviluppatore di app
Convalida le prestazioni dell'applicazione.	Avvia l'applicazione. Convalida la funzionalità e le prestazio ni dell'applicazione utilizzan do Automated Workload Repository (documentazione Oracle).	Sviluppatore di app, ingegnere , DevOps DBA

Risorse correlate

Riferimenti AWS

- Migrazione dei database Oracle sul cloud AWS
- Amazon EC2 per Oracle
- Migrazione di database Oracle voluminosi su AWS per ambienti multipiattaforma
- VPCs e sottoreti
- Tutorial: creare un VPC da utilizzare con un'istanza di database

Riferimenti Oracle

- Configurazioni Oracle Data Guard
- Importazione di Data Pump

Migra i sistemi RHEL BYOL verso istanze con licenza AWS inclusa utilizzando AWS MGN

Mike Kuznetsov, Amazon Web Services

Riepilogo

Quando migri i tuoi carichi di lavoro su AWS utilizzando AWS Application Migration Service (AWS MGN), potresti dover sollevare e spostare (reospitare) le istanze di Red Hat Enterprise Linux (RHEL) e modificare la licenza dal modello Bring Your Own License (BYOL) predefinito a un modello AWS License Included (LI) durante la migrazione. AWS MGN supporta un approccio scalabile che utilizza ID Amazon Machine Image (AMI). Questo modello descrive come effettuare la modifica della licenza sui server RHEL durante la migrazione del rehost su larga scala. Spiega inoltre come modificare la licenza per un sistema RHEL già in esecuzione su Amazon Elastic Compute Cloud (Amazon EC2) Elastic Compute Cloud (Amazon EC2).

Prerequisiti e limitazioni

Prerequisiti

- Accesso all'account AWS di destinazione
- AWS MGN inizializzato nell'account AWS e nella regione di destinazione per la migrazione (non richiesto se hai già effettuato la migrazione dal tuo sistema locale ad AWS)
- Un server RHEL di origine con una licenza RHEL valida

Architecture

Questo modello copre due scenari:

- Migrazione di un sistema da locale direttamente a un'istanza AWS LI utilizzando AWS MGN. Per questo scenario, segui le istruzioni del primo capitolo epico (Migrazione all'istanza LI - opzione 1) e del terzo capitolo.
- Modifica del modello di licenza da BYOL a LI per un sistema RHEL precedentemente migrato e già in esecuzione su Amazon EC2. Per questo scenario, segui le istruzioni contenute nella seconda pagina epica (Migrate to LI instance - opzione 2) e nella terza epic.

> **Note**
>
> La terza epopea riguarda la riconfigurazione della nuova istanza RHEL per utilizzare i server Red Hat Update Infrastructure (RHUI) forniti da AWS. Questo processo è lo stesso per entrambi gli scenari.

Tools (Strumenti)

Servizi AWS

- AWS Application Migration Service (AWS MGN) ti aiuta a reospitare (lift and shift) le applicazioni nel cloud AWS senza modifiche e con tempi di inattività minimi.

Epiche

Migrazione all'istanza LI - opzione 1 (per un sistema RHEL locale)

Operazione	Description	Competenze richieste
Trova l'ID AMI dell'istanza RHEL AWS LI nella regione di destinazione.	Visita AWS Marketplace o usa la console Amazon EC2 per trovare l'ID AMI RHEL che corrisponde alla versione del sistema di origine RHEL (ad esempio, RHEL-7.7) e annota l'ID AMI. Sulla console Amazon EC2, puoi filtrare AMIs utilizzando uno dei seguenti termini di ricerca: • Descrizione = Fornito da Red Hat, Inc. • Nome AMI = RHEL-7.7	Amministratore cloud
Configura le impostazioni di avvio di AWS MGN.	1. Sulla console AWS MGN, aggiungi il sistema RHEL di origine: installa	Amministratore del cloud

Operazione	Description	Competenze richieste
	l'agente di replica AWS e aggiungi il server di origine seguendo le istruzioni nella documentazione di AWS MGN. 2. Nella pagina Server di origine, scegli il sistema RHEL di origine, quindi scegli la scheda Impostazioni di avvio. 3. Nella sezione Impostazioni generali di avvio, scegli Modifica. Per disabilitare la selezione automatica e specificare manualmente il tipo di istanza di destinazione, modifica la dimensione corretta del tipo di istanza su Nessuno, quindi scegli Salva impostazioni. Ciò consente di utilizzare il tipo di istanza configurato nel modello di lancio di Amazon EC2. Per ulteriori informazioni, consulta la documentazione di AWS MGN. 4. Nella sezione EC2 Launch Template, scegli Modifica. Nella finestra di dialogo Informazioni sulla modifica dei modelli di avvio EC2, scegli nuovamente Modifica. Si apre la console Amazon EC2 in modo da	

Operazione	Description	Competenze richieste
	poter modificare il modello per questa istanza. 5. Consulta le considerazioni chiave nella documenta zione di AWS MGN. **Note** Puoi ignorare l'avvertimento di non scegliere la tua AMI. 6. Sulla console Amazon EC2, nel nuovo modello di avvio, modifica quanto segue: • Per AMI, specifica l'ID AMI che hai identificato in precedenza oppure cerca RHEL- x e specifica la versione richiesta (ad esempio, RHEL-7.7). • Per Tipo di istanza, impostate il tipo di istanza di destinazione desiderat o. • Lasciate invariate le seguenti sezioni: coppia di chiavi (login), impostazi oni di rete (a meno che non vogliate specificare una sottorete di destinazi one e gruppi di sicurezza), Storage, tag Resource	

Operazione	Description	Competenze richieste
	(a meno che non vogliate aggiungere o modificare tag). • (Facoltativo) Nella sezione Dettagli avanzati, specifica il ruolo del profilo dell'istanza IAM, se necessario per la gestione futura da parte di AWS Systems Manager. 7. Scegli Crea versione modello, quindi scegli il link nel messaggio di successo per visualizzare il modello di lancio. 8. Scegli Azioni, Imposta versione predefinita. Per Versione modello, seleziona la versione più recente (versione 2 per un nuovo sistema), quindi scegli Imposta come versione predefinita. AWS MGN utilizzerà ora questa versione del modello di lancio per avviare istanze di test o cutover. Per ulteriori informazioni, consulta la documentazione di AWS MGN.	

Operazione	Description	Competenze richieste
Convalida le impostazioni.	1. Nella console AWS MGN, nella pagina Server di origine, scegli il tuo server di origine, quindi scegli la scheda Impostazioni di avvio. 2. Nella sezione EC2 Launch Template, verifica che i parametri del tipo di istanza, della sottorete e dei gruppi di sicurezza siano impostati correttamente. **Note** Questa sezione non mostra l'ID AMI selezionato. Per visualizzare l'ID, puoi aprire la console Amazon EC2, la vista Launch Templates e cercare l'ID del modello mostrato in questa sezione.	Amministratore cloud

Operazione	Description	Competenze richieste
Avvia la nuova istanza LI.	1. Una volta completata la sincronizzazione iniziale, la colonna Migration lifecycle per il server nella pagina Source servers della console AWS MGN cambia in Ready for testing. Per avviare la nuova istanza di test, scegli il tuo server di origine, apri il menu Test and cutover, quindi scegli Launch test instances. Scegli Visualizza i dettagli del lavoro per monitorare lo stato del processo di lancio. Per ulteriori informazioni, consulta la documentazione di AWS MGN. 2. Attendi il completamento del processo di avvio, quindi apri la pagina dei dettagli dell'istanza EC2 lanciata. Scegli la scheda Dettagli e verifica che la sezione Dettagli dell'istanza contenga quanto segue: • Dettagli della piattaforma: «Red Hat Enterprise Linux» • Nome AMI: il nome dell'AMI che hai specificato nel modello di lancio EC2	Amministratore cloud

Operazione	Description	Competenze richieste
	3. Passa alla nuova istanza LI seguendo le istruzioni nella documentazione di AWS MGN. 4. Riconfigura la nuova istanza per utilizzare i server RHUI forniti da AWS seguendo i passaggi dell'ultima epopea.	

Migrazione a un'istanza LI - opzione 2 (per un'istanza RHEL BYOL EC2)

Operazione	Description	Competenze richieste
Migra la tua istanza RHEL BYOL EC2 su un'istanza AWS LI.	Puoi cambiare i sistemi RHEL precedentemente migrati su AWS come BYOL in istanze AWS LI spostando i relativi dischi (volumi Amazon Elastic Block Store) e collegandoli a una nuova istanza LI. Per effettuare questo passaggio, segui questi passaggi: 1. Avvia una nuova istanza RHEL di destinazione da un AMI RHEL LI. Assicurati che l'AMI che hai selezionato: • Utilizza la stessa versione di RHEL dell'istanza RHEL corrente. • Ha lo stesso processo di avvio (BIOS o UEFI)	Amministratore cloud

Operazione	Description	Competenze richieste
	dell'istanza RHEL corrente. Ad esempio, se il server di origine è basato su BIOS, utilizza l'AMI RHEL di AWS Marketplace anch'essa basata sul BIOS; per i sistemi basati su UEFI, scegli l'AMI basata su UEFI. 2. Arresta entrambe le istanze: la nuova istanza LI e l'istanza sorgente originale. 3. Scollega tutti i volumi EBS (incluso il disco principale) dalla nuova istanza LI ed eliminali. 4. Scollega tutti i volumi EBS (incluso il disco principal e) dalla vecchia istanza di origine e collegali alla nuova istanza LI. Mantieni la stessa mappatura dei volumi sui dispositivi. (Ad esempio, il volume EBS precedentemente collegato all'`/dev/sda`unità deve essere collegato come `/dev/sda` alla nuova istanza). 5. Eliminare l'istanza di origine (ora senza disco). 6. Avviate la nuova istanza LI. Accedi all'istanza e	

Operazione	Description	Competenze richieste
	riconfigurala per utilizzare i server RHUI forniti da AWS seguendo i passaggi della prossima epic.	

Riconfigurazione del sistema operativo RHEL per utilizzare RHUI fornito da AWS: entrambe le opzioni

Operazione	Description	Competenze richieste
Annulla la registrazione del sistema operativo dalla sottoscrizione e dalla licenza Red Hat.	Dopo la migrazione e il completamento con successo, il sistema RHEL deve essere rimosso dalla sottoscrizione Red Hat per interrompere l'utilizzo della licenza Red Hat ed evitare una doppia fatturazione. Per rimuovere RHEL OS dalla sottoscrizione Red Hat, seguite la procedura descritta nella documentazione di Red Hat Subscription Management (RHSM). Utilizzare il comando CLI : `subscription-manager unregister` Puoi anche disabilitare il plugin di gestione delle sottoscrizioni per interrompere il controllo dello stato dell'abbonamento a ogni chiamata yum.	Linux o amministratore di sistema

Operazione	Description	Competenze richieste
	Per fare ciò, modifica il file di configurazione `/etc/yum/pluginconf.d/subscription-manager.conf` e modifica il parametro `enabled=1` in `enabled=0`.	

Operazione	Description	Competenze richieste
Sostituisci la vecchia configurazione di aggiornamento (RHUI, rete Red Hat Satellite, repository yum) con la RHUI fornita da AWS.	È necessario riconfigurare il sistema RHEL migrato per utilizzare i server RHUI forniti da AWS. Ciò consente di accedere ai server RHUI all'interno delle regioni AWS senza richiedere l'infrastruttura di aggiornamento esterna. La modifica prevede il seguente processo: 1. Esegui il backup della configurazione yum esistente. 2. Rimuovi la vecchia configurazione e i pacchetti RHUI (yum repositories). 3. Aggiungi i nuovi pacchetti di configurazione e certificati RHUI forniti da AWS. È necessario recuperarli da un'altra istanza RHEL su AWS perché questi pacchetti di configurazione sono disponibili solo sui server RHUI forniti da AWS. Ecco i passaggi e i comandi dettagliati: 1. Esegui il backup della configurazione e dei certificati yum esistenti copiando tutte `/etc/yum*`	Linux o amministratore di sistema

Operazione	Description	Competenze richieste

	le `/etc/pki/*` cartelle in una posizione di backup. Esempio:
	`mkdir yum-backup`
	`cp -ra /etc/yum* /etc/pki ./yum-backup`
	`tar czf yum-backu p.tgz ./yum-backup`
	2. Rimuovi la vecchia configurazione e i pacchetti RHUI:
	a. Trova tutti i pacchetti RHUI installati:
	`sudo rpm -qa | grep rhui`
	b. Eliminate questi pacchetti:
	`sudo yum remove $(rpm -qa | grep rhui)`
	c. Rimuovi il `/etc/yum/vars/releasever` file, se esiste.
	3. Aggiungi i nuovi pacchetti RHUI e certificati forniti da AWS. È necessario recuperarli da un'altra istanza RHEL su AWS. Esistono vari modi per eseguire questa operazion e. Ad esempio, puoi	

Operazione	Description	Competenze richieste

	seguire le istruzioni fornite nell'articolo della Red Hat Knowledgebase:
	a. Avvia un'altra istanza RHEL (RHEL-EC2) da AWS Marketplace.
	b. Scarica due pacchetti da questa istanza: l'ultimo pacchetto di configura zione del client RHUI e i certificati dell'autorità di certificazione (CA). Ad esempio, esegui questo comando dal desktop:
	`ssh RHEL-EC2 "sudo yumdownloader ca-certificates rh-amazon-rhui-cli ent"`
	c. Copia i pacchetti dall'ista nza RHEL-EC2 al nuovo sistema migrato. Esempio:
	`scp RHEL-EC2:rh-amazon-rhui-cli ent\* RHEL-EC2:ca-certificates\* . ssh <migrated-instance> "mkdir /tmp/amazon" scp rh-amazon-rhui-client* ca-certif icates* <migrated`	

Operazione	Description	Competenze richieste

	`-instance>:/tmp/am azon`
	d. Installa i nuovi pacchetti di configurazione RHUI e CA sull'istanza migrata:
	`ssh <migrated-instance> "sudo rpm -Uhv /tmp/amazon/*"`	

Convalida la configurazione.	Nell'istanza di destinazione migrata, verifica che la nuova configurazione sia corretta:
	`sudo yum clean all`
	`sudo yum repolist`
		Linux o amministratore di sistema

Risorse correlate

- Guida per l'utente di AWS Application Migration Service (AWS MGN)
- Ottieni il supporto di un pacchetto client AWS RHUI IMDSv2 (articolo della Red Hat Knowledgebase)
- Modelli di lancio di Amazon EC2 (documentazione Amazon EC2)

Esegui la migrazione di un database Microsoft SQL Server locale su Amazon EC2

Senthil Ramasamy, Amazon Web Services

Riepilogo

Questo modello descrive come migrare un database Microsoft SQL Server locale a Microsoft SQL Server su un'istanza Amazon Elastic Compute Cloud (Amazon EC2). Include due opzioni per la migrazione: utilizzando AWS Database Migration Service (AWS DMS) o utilizzando strumenti nativi di Microsoft SQL Server come backup e ripristino, Copy Database Wizard o copia e collega il database.

Prerequisiti e limitazioni

Prerequisiti

- Un account attivo AWS
- Un sistema operativo supportato da Amazon EC2 (per un elenco completo delle versioni del sistema operativo supportate, consulta Amazon EC2 FAQs)
- Un database di origine Microsoft SQL Server in un data center locale

Versioni del prodotto

- Per i database locali e di EC2 istanze Amazon, AWS DMS supporta:
  - Versioni di SQL Server 2005, 2008, 2008R2, 2012, 2014, 2016, 2017 e 2019
  - Edizioni Enterprise, Standard, Workgroup, Developer e Web
- Per l'elenco più recente delle versioni supportate, vedere Utilizzo di un database Microsoft SQL Server come destinazione per AWS DMS.

Architecture

Stack tecnologico di origine

- Database Microsoft SQL Server locale

Stack tecnologico Target

- Database Microsoft SQL Server su un' EC2 istanza

Architettura Target

Architettura di migrazione dei dati

- Usando AWS DMS

- Utilizzo di strumenti nativi di SQL Server

Tools (Strumenti)

- AWS Database Migration Service (AWS DMS) consente di migrare i dati da e verso database commerciali e open source ampiamente utilizzati, tra cui Oracle, SQL Server, MySQL e PostgreSQL. Puoi utilizzarli AWS DMS per migrare i dati in Cloud AWS, tra istanze locali (tramite una Cloud AWS configurazione) o tra combinazioni di configurazioni cloud e locali.
- AWS Schema Conversion Tool (AWS SCT) supporta migrazioni di database eterogenee convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione.
- Gli strumenti nativi di Microsoft SQL Server includono backup e ripristino, Copia guidata del database e copia e allega database.

Epiche

Pianifica la migrazione

Operazione	Description	Competenze richieste
Convalida le versioni del database di origine e di destinazione.		DBA
Identifica la versione del sistema operativo di destinazione.		DBA, amministratore di sistema

Operazione	Description	Competenze richieste
Identifica i requisiti hardware per l'istanza del server di destinazione in base all'elenco di compatibilità e ai requisiti di capacità di Microsoft SQL Server.		DBA, amministratore di sistema
Identifica i requisiti di archiviazione per tipo e capacità.		DBA, amministratore di sistema
Identifica i requisiti di rete, tra cui latenza e larghezza di banda.		DBA, amministratore di sistema
Scegli il tipo di EC2 istanza in base alla capacità, alle funzionalità di archiviazione e alle funzionalità di rete.		DBA, amministratore di sistema
Identifica i requisiti di sicurezza dell'accesso alla rete e all'host per i database di origine e di destinazione.		DBA, amministratore di sistema
Identificare un elenco di utenti necessari per l'installazione del software Microsoft SQL Server.		DBA, amministratore di sistema
Determinare la strategia di backup.		DBA
Determinare i requisiti di disponibilità.		DBA

Operazione	Description	Competenze richieste
Identifica la migrazione delle applicazioni e la strategia di cutover.		DBA, amministratore di sistema

Configura l'infrastruttura

Operazione	Description	Competenze richieste
Crea un cloud privato virtuale (VPC) e sottoreti.		Amministratore di sistema
Crea gruppi di sicurezza e una lista di controllo degli accessi alla rete (ACL).		Amministratore di sistema
Configura e avvia un' EC2 istanza.		Amministratore di sistema

Installa il software

Operazione	Description	Competenze richieste
Creare gli utenti e i gruppi necessari per il software Microsoft SQL Server.		DBA, amministratore di sistema
Scarica il software Microsoft SQL Server.		DBA, amministratore di sistema
Installa il software Microsoft SQL Server sull' EC2 istanza e configura il server.		DBA, amministratore di sistema

Migrazione dei dati - opzione 1

Operazione	Description	Competenze richieste
Utilizza strumenti nativi di Microsoft SQL Server o strumenti di terze parti per migrare gli oggetti e i dati del database.	Gli strumenti includono backup e ripristino, Copy Database Wizard e copia e allega il database. Per ulteriori informazioni, vedere la guida Migrazione dei database di Microsoft SQL Server a. Cloud AWS	DBA

Migrazione dei dati - opzione 2

Operazione	Description	Competenze richieste
Esegui la migrazione dei dati utilizzando AWS DMS.	Per ulteriori informazioni sull'utilizzo AWS DMS, consulta i collegamenti nella sezione Risorse correlate.	DBA

Migrare l'applicazione

Operazione	Description	Competenze richieste
Segui la strategia di migrazion e delle applicazioni.	Utilizzate AWS Schema Conversion Tool (AWS SCT) per analizzare e modificar e il codice SQL incorporato nel codice sorgente dell'appl icazione.	DBA, proprietario dell'app

Tagliare

Operazione	Description	Competenze richieste
Segui la strategia di commutazione delle applicazioni.		DBA, proprietario dell'app, amministratore di sistema

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi tutte le AWS risorse temporanee.	Le risorse temporanee includono l'istanza AWS DMS di replica e l' EC2 istanza for AWS SCT.	DBA, amministratore di sistema
Rivedi e convalida i documenti del progetto.		DBA, proprietario dell'app, amministratore di sistema
Raccogli le metriche necessarie per la migrazione, la percentuale di risparmio sui costi manuali rispetto a quelli relativi agli strumenti e così via.		DBA, proprietario dell'app, amministratore di sistema
Chiudi il progetto e fornisci feedback.		DBA, proprietario dell'app, amministratore di sistema

Risorse correlate

Riferimenti

- Migrazione dei database Microsoft SQL Server verso Cloud AWS
- Amazon EC2
- Amazon EC2 FAQs

- EC2 Prezzi Amazon
- AWS Database Migration Service
- Prodotti Microsoft su AWS
- Licenze Microsoft su AWS
- Microsoft SQL Server su AWS

Tutorial e video

- Guida introduttiva ad Amazon EC2
- Nozioni di base su AWS Database Migration Service
- Unisci un' EC2 istanza Amazon al tuo Simple AD Active Directory
- Unisci un' EC2 istanza Amazon alla tua AWS Managed Microsoft AD Active Directory
- AWS Database Migration Service(video)
- Introduzione ad Amazon EC2 — Elastic Cloud Server & Hosting con AWS (video)

Rehosting dei carichi di lavoro locali nel cloud AWS: lista di controllo per la migrazione

Srikanth Rangavajhala, Amazon Web Services

Riepilogo

Il rehosting dei carichi di lavoro locali nel cloud Amazon Web Services (AWS) prevede le seguenti fasi di migrazione: pianificazione, pre-discovery, discovery, build, test e cutover. Questo modello delinea le fasi e le attività correlate. Le attività sono descritte a un livello elevato e supportano circa il 75% di tutti i carichi di lavoro delle applicazioni. È possibile implementare queste attività nell'arco di due o tre settimane in un ciclo di sprint agile.

È necessario esaminare e verificare queste attività con il team di migrazione e i consulenti. Dopo la revisione, è possibile raccogliere gli input, eliminare o rivalutare le attività necessarie per soddisfare i requisiti e modificare altre attività per supportare almeno il 75% dei carichi di lavoro delle applicazioni nel portafoglio. È quindi possibile utilizzare uno strumento di gestione dei progetti agile come Atlassian Jira o Rally Software per importare le attività, assegnarle alle risorse e tenere traccia delle attività di migrazione.

Lo schema presuppone che tu stia utilizzando AWS Cloud Migration Factory per riospitare i tuoi carichi di lavoro, ma puoi usare il tuo strumento di migrazione preferito.

Amazon Macie può aiutarti a identificare i dati sensibili nelle tue knowledge base, archiviati come fonti di dati, log di invocazione di modelli e archivi di richieste nei bucket Amazon Simple Storage Service (Amazon S3). Per ulteriori informazioni, consulta la documentazione di Macie.

Prerequisiti e limitazioni

Prerequisiti

- Strumento di gestione dei progetti per tenere traccia delle attività di migrazione (ad esempio, Atlassian Jira o Rally Software)
- Strumento di migrazione per il rehosting dei carichi di lavoro su AWS (ad esempio, Cloud Migration Factory)

Architecture

Piattaforma di origine

- Stack di sorgenti locale (incluse tecnologie, applicazioni, database e infrastruttura)

Piattaforma di destinazione

- Stack di obiettivi AWS Cloud (tra cui tecnologie, applicazioni, database e infrastruttura)

Architettura

Il diagramma seguente illustra il rehosting (scoperta e migrazione di server da un ambiente di origine locale ad AWS) utilizzando Cloud Migration Factory e AWS Application Migration Service.

Tools (Strumenti)

- Puoi utilizzare uno strumento di migrazione e gestione dei progetti a tua scelta.

Epiche

Fase di pianificazione

Operazione	Description	Competenze richieste
Gestisci il backlog precedente alla scoperta.	Conduci la sessione di lavoro sulla gestione del backlog pre-discovery con i responsabili di reparto e i proprietari delle applicazioni.	Responsabile del progetto, Agile Scrum Leader
Conduci la sessione di lavoro sulla pianificazione dello sprint.	Come esercizio di analisi, distribuisci le applicazioni che desideri migrare tra sprint e wave.	Responsabile di progetto, Agile Scrum Leader

Fase di pre-scoperta

Operazione	Description	Competenze richieste
Conferma la conoscenza dell'applicazione.	Conferma e documenta il proprietario dell'applicazione	Specialista in migrazione (intervistatore)

Operazione	Description	Competenze richieste
	e la sua conoscenza dell'appl icazione. Stabilisci se esiste un'altra persona di riferimento per le domande tecniche.	
Determinare i requisiti di conformità delle applicazioni.	Verifica con il proprietario dell'applicazione che l'applica zione non deve soddisfar e i requisiti per il Payment Card Industry Data Security Standard (PCI DSS), il Sarbanes-Oxley Act (SOX), le informazioni di identific azione personale (PII) o altri standard. Se esistono requisiti di conformità, i team devono completare i controlli di conformità sui server che verranno migrati.	Specialista in migrazione (intervistatore)
Conferma i requisiti di rilascio della versione di produzione.	Confermate i requisiti per il rilascio dell'applicazione migrata in produzione (come la data di rilascio e la durata del periodo di inattività) con il proprietario dell'applicazione o il contatto tecnico.	Specialista in migrazione (intervistatore)
Ottieni l'elenco dei server.	Ottieni l'elenco dei server associati all'applicazione di destinazione.	Specialista in migrazione (intervistatore)
Ottieni il diagramma logico che mostra lo stato attuale.	Ottenete il diagramma dello stato corrente dell'applicazione dall'architetto aziendale o dal proprietario dell'applicazione.	Specialista in migrazione (intervistatore)

Operazione	Description	Competenze richieste
Crea un diagramma logico che mostri lo stato di destinazione.	Crea un diagramma logico dell'applicazione che mostri l'architettura di destinazione su AWS. Questo diagramma dovrebbe illustrare i server, la connettività e i fattori di mappatura.	Architetto aziendale, imprenditore
Ottieni informazioni sul server.	Raccogli informazioni sui server associati all'applicazione, inclusi i dettagli di configurazione.	Specialista in migrazione (intervistatore)
Aggiungi informazioni sul server al modello di scoperta.	Aggiungere informazioni dettagliate sul server al modello di rilevamento dell'applicazione (vedere `mobilize-application-questionnaire.xlsx` nell'allegato per questo modello). Questo modello include tutti i dettagli relativi alla sicurezza, all'infrastruttura, al sistema operativo e alla rete relativi all'applicazione.	Specialista in migrazione (intervistatore)
Pubblica il modello di rilevamento delle applicazioni.	Condividi il modello di rilevamento delle applicazioni con il proprietario dell'applicazione e il team di migrazione per un accesso e un utilizzo comuni.	Specialista in migrazione (intervistatore)

Fase di scoperta

Operazione	Description	Competenze richieste
Conferma l'elenco dei server.	Confermate l'elenco dei server e lo scopo di ciascun server con il proprietario dell'appl icazione o il responsabile tecnico.	Specialista in migrazione
Identifica e aggiungi gruppi di server.	Identifica gruppi di server come server Web o server di applicazioni e aggiungi queste informazioni al modello di rilevamento delle applicazioni. Seleziona il livello dell'appl icazione (web, applicazi one, database) a cui deve appartenere ogni server.	Specialista in migrazione
Compila il modello di scoperta dell'applicazione.	Completa i dettagli del modello di rilevamento delle applicazi oni con l'aiuto del team di migrazione, del team dell'appl icazione e di AWS.	Specialista in migrazione
Aggiungi i dettagli mancanti del server (team del middlewar e e del sistema operativo).	Chiedi ai team del middlewar e e del sistema operativo (OS) di esaminare il modello di individuazione delle applicazi oni e aggiungere eventuali dettagli mancanti sul server, incluse le informazioni sul database.	Specialista in migrazione
Ottieni le regole inbound/o utbound del traffico (team di rete).	Chiedi al team di rete di ottenere le regole inbound/ outbound del traffico per i	Specialista in migrazione

Operazione	Description	Competenze richieste
	server di origine e di destinazione. Il team di rete deve inoltre aggiungere le regole firewall esistenti, esportarle in un formato di gruppo di sicurezza e aggiungere i sistemi di bilanciamento del carico esistenti al modello di rilevamento delle applicazioni.	
Identifica i tag richiesti.	Determina i requisiti di etichettatura per l'applicazione.	Specialista in migrazione
Crea i dettagli della richiesta del firewall.	Acquisisci e filtra le regole del firewall necessarie per comunicare con l'applicazione.	Specialista in migrazione, architetto di soluzioni, responsabile della rete
Aggiorna il tipo di EC2 istanza.	Aggiorna il tipo di istanza Amazon Elastic Compute Cloud (Amazon EC2) da utilizzare nell'ambiente di destinazione, in base ai requisiti dell'infrastruttura e del server.	Specialista in migrazione, architetto di soluzioni, responsabile della rete
Identifica il diagramma dello stato attuale.	Identifica o crea il diagramma che mostra lo stato corrente dell'applicazione. Questo diagramma verrà utilizzato nella richiesta di sicurezza delle informazioni (InfoSec).	Specialista in migrazione, Solutions architect

Operazione	Description	Competenze richieste
Finalizza il diagramma degli stati futuri.	Finalizza il diagramma che mostra lo stato futuro (di destinazione) dell'appl icazione. Questo diagramma verrà utilizzato anche nella richiesta. InfoSec	Specialista in migrazione, Solutions architect
Crea richieste di assistenz a per firewall o gruppi di sicurezza.	Crea richieste di servizi per firewall o gruppi di sicurezza (per sviluppo/QA, preproduz ione e produzione). Se utilizzi Cloud Migration Factory, includi porte specifiche per la replica se non sono già aperte.	Specialista in migrazion e, architetto di soluzioni, responsabile della rete
Esamina le richieste del firewall o del gruppo di sicurezza (InfoSec team).	In questo passaggio, il InfoSec team esamina e approva le richieste del firewall o del gruppo di sicurezza create nel passaggio precedente.	InfoSec ingegnere, specialista in migrazione
Implementa le richieste dei gruppi di sicurezza del firewall (team di rete).	Dopo che il InfoSec team ha approvato le richieste del firewall, il team di rete implementa le regole inbound/ outbound firewall richieste.	Specialista in migrazion e, architetto di soluzioni, responsabile della rete

Fase di costruzione (ripetere per ambienti di sviluppo/QA, preproduzione e produzione)

Operazione	Description	Competenze richieste
Importa i dati dell'applicazione e del server.	1. Verificate di aver effettuat o l'accesso al server di esecuzione della migrazion	Specialista in migrazione, amministratore cloud

Operazione	Description	Competenze richieste
	e come utente di dominio con autorizzazioni di amministratore locale sui server di origine interessati. 2. Utilizza il modulo di richiesta di migrazione per importare gli attributi per i server di origine interessa ti. Per ulteriori informazi oni, consulta la Guida all'implementazione di Cloud Migration Factory. Se non utilizzi Cloud Migration Factory, segui le istruzioni per configurare lo strumento di migrazione.	
Verifica i prerequisiti per i server di origine.	Connect con i server di origine inclusi nell'ambito per verificare prerequisiti come la porta TCP 1500, la porta TCP 443, lo spazio libero nel volume root, la versione.NET Framework e altri parametri. Questi sono necessari per la replica. Per ulteriori informazi oni, consulta la Guida all'imple mentazione di Cloud Migration Factory.	Specialista in migrazione, amministratore cloud

Operazione	Description	Competenze richieste
Crea una richiesta di servizio per installare gli agenti di replica.	Crea una richiesta di servizio per installare gli agenti di replica sui server interessa ti per lo sviluppo/QA, la preproduzione o la produzion e.	Specialista in migrazione, amministratore cloud
Installa gli agenti di replica.	Installa gli agenti di replica sui server di origine pertinenti sui computer di sviluppo/QA, di preproduzione o di produzion e. Per ulteriori informazioni, consulta la Guida all'imple mentazione di Cloud Migration Factory.	Specialista in migrazione, amministratore cloud
Invia gli script post-lancio.	Application Migration Service supporta gli script post-lancio per aiutarti ad automatizzare le attività a livello di sistema operativo, come l'install azione o la disinstallazione del software dopo l'avvio delle istanze di destinazione. Questo passaggio invia gli script post-avvio ai computer Windows o Linux, a seconda dei server identificati per la migrazione. Per istruzioni, consulta la Guida all'imple mentazione di Cloud Migration Factory.	Specialista in migrazione, amministratore cloud

Operazione	Description	Competenze richieste
Verifica lo stato della replica.	Conferma automaticamente lo stato di replica per i server di origine pertinenti utilizzan do lo script fornito. Lo script si ripete ogni cinque minuti fino a quando lo stato di tutti i server di origine nell'onda ta data diventa Healthy. Per istruzioni, consulta la Guida all'implementazione di Cloud Migration Factory.	Specialista in migrazione, amministratore cloud
Crea l'utente amministratore.	Potrebbe essere necessario un amministratore locale o un utente sudo sui computer di origine per risolvere eventuali problemi dopo l'interruzione della migrazione dai server di origine interessati ad AWS. Il team di migrazione utilizza questo utente per accedere al server di destinazione quando il server di autentica zione (ad esempio, il server DC o LDAP) non è raggiungi bile. Per istruzioni su questo passaggio, consulta la Guida all'implementazione di Cloud Migration Factory.	Specialista della migrazione, amministratore del cloud

Operazione	Description	Competenze richieste
Convalida il modello di lancio.	Convalida i metadati del server per assicurarti che funzionino correttamente e non contengano dati non validi. Questo passaggio convalida sia i metadati di test che quelli di cutover. Per istruzioni, consulta la Guida all'implementazione di Cloud Migration Factory.	Specialista in migrazione, amministratore cloud

Fase di test (ripetizione per ambienti di sviluppo/QA, preproduzione e produzione)

Operazione	Description	Competenze richieste
Crea una richiesta di assistenza.	Crea una richiesta di servizio per il team dell'infrastruttura e gli altri team per eseguire il trasferimento delle applicazioni alle istanze di sviluppo/QA, di pre-produzione o di produzione.	Specialista in migrazione, amministratore cloud
Configura un sistema di bilanciamento del carico (opzionale).	Configura i load balancer richiesti, come un Application Load Balancer o un load balancer F5 con iRules.	Specialista in migrazione, amministratore cloud
Avvia istanze per i test.	Avvia tutte le macchine di destinazione per una determinata ondata in Application Migration Service in modalità test. Per ulteriori informazioni, consulta la Guida	Specialista della migrazione, amministratore del cloud

Operazione	Description	Competenze richieste
	all'implementazione di Cloud Migration Factory.	
Verifica lo stato dell'istanza di destinazione.	Verifica lo stato dell'istanza di destinazione controlla ndo il processo di avvio di tutti i server di origine inclusi nell'ambito della stessa ondata. L'avvio delle istanze di destinazione può richiedere fino a 30 minuti. Puoi controlla re lo stato manualmente accedendo alla EC2 console Amazon, cercando il nome del server di origine e rivedendo la colonna Controllo dello stato. I controlli di stato 2/2 superati indicano che l'istanza è integra dal punto di vista dell'infrastruttura.	Specialista in migrazione, amministratore del cloud

Operazione	Description	Competenze richieste
Modifica le voci DNS.	Modifica le voci del Domain Name System (DNS). (Utilizzare `resolv.conf` o `host.conf` per un ambiente Microsoft Windows). Configura ogni EC2 istanza in modo che punti al nuovo indirizzo IP di questo host. **Note** Assicurati che non vi siano conflitti DNS tra i server locali e i server cloud AWS. Questo passaggio e i passaggi seguenti sono facoltativi, a seconda dell'ambiente in cui è ospitato il server.	Specialista in migrazione, amministratore del cloud
Verifica la connettività agli host di backend a partire dalle EC2 istanze.	Controlla gli accessi utilizzando le credenziali di dominio per i server migrati.	Specialista in migrazione, amministratore cloud
Aggiorna il record DNS A.	Aggiorna il record DNS A per ogni host in modo che punti al nuovo indirizzo IP EC2 privato di Amazon.	Specialista in migrazione, amministratore cloud
Aggiorna il record DNS CNAME.	Aggiorna il record DNS CNAME per i nomi virtuali IPs (load balancer) in modo che punti al cluster per i server Web e applicativi.	Specialista in migrazione, amministratore cloud

Operazione	Description	Competenze richieste
Testa l'applicazione negli ambienti applicabili.	Accedi alla nuova EC2 istanza e testa l'applicazione negli ambienti di sviluppo/QA, di preproduzione e produzione.	Specialista in migrazione, amministratore cloud
Contrassegna come pronto per il cutover.	Al termine del test, modifica lo stato del server di origine per indicare che è pronto per il cutover, in modo che gli utenti possano avviare un'istanza cutover. Per istruzioni, consulta la Guida all'implementazione di Cloud Migration Factory.	Specialista in migrazione, amministratore cloud

Fase di cutover

Operazione	Description	Competenze richieste
Crea un piano di distribuzione della produzione.	Crea un piano di distribuzione della produzione (incluso un piano di backout).	Specialista in migrazione, amministratore cloud
Notifica al team operativo i tempi di inattività.	Notifica al team operativo la pianificazione dei tempi di inattività dei server. Alcuni team potrebbero richiedere un ticket di richiesta di modifica o di richiesta di servizio (CR/SR) per questa notifica.	Specialista in migrazione, amministratore cloud
Replica le macchine di produzione.	Replica le macchine di produzione utilizzando Application Migration Service	Specialista in migrazione, amministratore cloud

Operazione	Description	Competenze richieste
	o un altro strumento di migrazione.	
Chiudi i server di origine pertinenti.	Dopo aver verificato lo stato di replica dei server di origine, è possibile spegnere i server di origine per interrompere le transazioni dalle applicazioni client ai server. È possibile spegnere i server di origine nella finestra di dialogo. Per ulteriori informazioni, consulta la Guida all'implementazione di Cloud Migration Factory.	Amministratore cloud
Avvia istanze per cutover.	Avvia tutte le macchine di destinazione per una determinata ondata in Application Migration Service in modalità cutover. Per ulteriori informazioni, consulta la Guida all'implementazione di Cloud Migration Factory.	Specialista in migrazione, amministratore cloud
Recupera l'istanza IPs di destinazione.	Recuperate le istanze IPs di destinazione. Se l'aggiorn amento DNS è un processo manuale nel tuo ambiente, dovrai ottenere i nuovi indirizzi IP per tutte le istanze di destinazione. Per ulteriori informazioni, consulta la Guida all'implementazione di Cloud Migration Factory.	Specialista in migrazione, amministratore cloud

Operazione	Description	Competenze richieste
Verifica le connessioni al server di destinazione.	Dopo aver aggiornato i record DNS, connettiti alle istanze di destinazione con il nome host per verificare le connessioni. Per ulteriori informazioni, consulta la Guida all'implementazione di Cloud Migration Factory.	Specialista in migrazione, amministratore cloud

Risorse correlate

- Come migrare
- Guida all'implementazione di AWS Cloud Migration Factory
- Automatizzazione delle migrazioni di server su larga scala con Cloud Migration Factory
- Guida per l'utente di AWS Application Migration Service
- Programma di accelerazione della migrazione AWS

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Configura un'infrastruttura Multi-AZ per SQL Server Always On FCI utilizzando Amazon FSx

Manish Garg, T.V.R.L.Phani Kumar Dadi, Nishad Mankar e RAJNEESH TYAGI, Amazon Web Services

Riepilogo

Se è necessario migrare rapidamente un gran numero di istanze di Microsoft SQL Server Always On Failover Cluster (FCIs), questo modello può aiutare a ridurre al minimo i tempi di provisioning. Utilizzando l'automazione e Amazon FSx for Windows File Server, riduce gli sforzi manuali, gli errori causati dall'uomo e il tempo necessario per implementare un gran numero di cluster.

Questo modello configura l'infrastruttura per SQL Server FCIs in una distribuzione Multi-Availability Zone (Multi-AZ) su Amazon Web Services (AWS). Il provisioning dei servizi AWS necessari per questa infrastruttura è automatizzato utilizzando CloudFormation modelli AWS. L'installazione di SQL Server e la creazione di nodi di cluster su un'istanza Amazon Elastic Compute Cloud (Amazon EC2) vengono eseguite utilizzando PowerShell comandi.

Questa soluzione utilizza un file system Multi-AZ Amazon FSx for Windows ad alta disponibilità come testimone condiviso per l'archiviazione dei file di database SQL Server. Il FSx file system Amazon e le istanze EC2 Windows che ospitano SQL Server sono uniti allo stesso dominio AWS Directory Service for Microsoft Active Directory (AWS Managed Microsoft AD).

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un utente AWS con autorizzazioni sufficienti per effettuare il provisioning di risorse utilizzando modelli AWS CloudFormation
- AWS Directory Service per Microsoft Active Directory
- Credenziali in AWS Secrets Manager per l'autenticazione su AWS Managed Microsoft AD in una coppia chiave-valore:
  - `ADDomainName`: <Domain Name>
  - `ADDomainJoinUserName`: <Domain Username>
  - `ADDomainJoinPassword`: <Domain User Password>
  - `TargetOU`: <Target OU Value>

> **Note**
>
> Utilizzerai lo stesso nome chiave nell'automazione di AWS Systems Manager per l'attività di join di AWS Managed Microsoft AD.

- File multimediali di SQL Server per l'installazione di SQL Server e creazione di account di servizio o dominio Windows, che verranno utilizzati durante la creazione del cluster
- Un cloud privato virtuale (VPC), con due sottoreti pubbliche in zone di disponibilità separate, due sottoreti private nelle zone di disponibilità, un gateway Internet, gateway NAT, associazioni di tabelle di routing e un jump server

Versioni del prodotto

- Windows Server 2012 R2 e Microsoft SQL Server 2016

Architecture

Stack tecnologico di origine

- SQL Server locale con FCIs utilizzo di un'unità condivisa

Stack tecnologico Target

- EC2 Istanze AWS
- File server Amazon FSx per Windows
- Guida introduttiva di AWS Systems Manager Automation
- Configurazioni di rete (VPC, sottoreti, gateway Internet, gateway NAT, jump server, gruppi di sicurezza)
- AWS Secrets Manager
- AWS Managed Microsoft AD
- Amazon EventBridge
- AWS Identity and Access Management (IAM)

Architettura Target

Il diagramma seguente mostra un account AWS in una singola regione AWS, con un VPC che include due zone di disponibilità, due sottoreti pubbliche con gateway NAT, un jump server nella prima sottorete pubblica, due sottoreti private, ciascuna con EC2 un'istanza per un nodo SQL Server in un gruppo di sicurezza dei nodi e un file system FSx Amazon che si connette a ciascuno dei nodi SQL Server. Sono inclusi anche AWS Directory Service EventBridge, Amazon, AWS Secrets Manager e AWS Systems Manager.

Automazione e scalabilità

- Puoi utilizzare AWS Systems Manager per unirti ad AWS Managed Microsoft AD ed eseguire l'installazione di SQL Server.

Tools (Strumenti)

Servizi AWS

- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account e regioni AWS.
- AWS Directory Service offre diversi modi per utilizzare Microsoft Active Directory (AD) con altri servizi AWS come Amazon Elastic Compute Cloud (Amazon EC2), Amazon Relational Database Service (Amazon RDS) per SQL Server e FSx Amazon per Windows File Server.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, funzioni AWS Lambda, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altri account AWS.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Secrets Manager ti aiuta a sostituire le credenziali codificate nel codice, comprese le password, con una chiamata API a Secrets Manager per recuperare il segreto a livello di codice.
- AWS Systems Manager ti aiuta a gestire le applicazioni e l'infrastruttura in esecuzione nel cloud AWS. Semplifica la gestione delle applicazioni e delle risorse, riduce i tempi di rilevamento e risoluzione dei problemi operativi e ti aiuta a gestire le tue risorse AWS in modo sicuro su larga scala.

Altri strumenti

- PowerShellè un programma di gestione dell'automazione e della configurazione di Microsoft che funziona su Windows, Linux e macOS. Questo modello utilizza PowerShell script.

Deposito di codice

Il codice per questo pattern è disponibile nel repository GitHub aws-windows-failover-cluster-automation.

Best practice

- I ruoli IAM utilizzati per implementare questa soluzione devono rispettare il principio del privilegio minimo. Per ulteriori informazioni, consulta la documentazione di IAM.
- Segui le CloudFormation best practice di AWS.

Epiche

Implementa l'infrastruttura

Operazione	Description	Competenze richieste
Implementa lo CloudFormation stack Systems Manager.	1. Accedi al tuo account AWS e apri la Console di gestione AWS. 2. Accedere alla CloudFormation console e creare lo CloudFormation stack Systems Manager caricando il `ssm.yaml` modello. Fornite i valori per i seguenti parametri: • StateUnJoinAssociationLoggingBucketName— Fornisci un nome per il bucket S3 che il	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	modello creerà per scopi di registrazione. • SSMAssociationADUnjoinNome: fornisci un nome per la risorsa. `AWS::SSM::Association` • SSMAutomationDocumentName— Fornire un nome per il runbook Systems Manager Automation. • EventBridgeName— Fornire un nome per il bus dell' EventBridge evento. 3. Implementa lo CloudFormation stack Systems Manager avviando il modello. `ssm.yaml` CloudFormation Il modello creerà il runbook Systems Manager Automation che viene avviato all'avvio di una nuova EC2 istanza con il tag. `ADJoined: FSXADD` L'Automation runbook aggiungerà l'istanza alla directory AWS Managed Microsoft AD.	

Operazione	Description	Competenze richieste
Implementa lo stack di infrastruttura.	Dopo una corretta implement azione dello stack Systems Manager, crea lo `infra` stack, che include nodi di EC2 istanza, gruppi di sicurezza, il file system Amazon FSx for Windows File Server e il ruolo IAM. 1. Passa alla CloudFormation console e avvia il `infra-cf.yaml` modello. Per distribuire questo stack, sono necessari i seguenti parametri: • `ActiveDirectoryId` — ID per AWS Managed Microsoft AD • `ADDnsIpAddresses1` — Indirizzo IP DNS primario di AWS Managed Microsoft AD • `ADDnsIpAddresses2` — Indirizzo IP DNS secondario di AWS Managed Microsoft AD • `FSxSecurityGroupName` — Nome del gruppo di FSx sicurezza Amazon • `FSxWindowsFileSystemName` — Nome dell' FSx unità Amazon	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	• `ImageID`— ID dell'immagine Windows 2012 R2 di base o Amazon Machine Image (AMI) utilizzata per creare il nodo dell'istanza di SQL Server • `KeyPairName` — Coppia chiave-valore da collegare ai nodi dell' EC2 istanza per l'accesso • `Node1SecurityGroupName` — Nome del primo gruppo di sicurezza del nodo • `Node2SecurityGroupName` — Nome del gruppo di sicurezza del secondo nodo • `OUSecretName` — Nome del segreto che contiene le informazioni di AWS Managed Microsoft AD • `PrivateSubnet1` — ID della prima sottorete privata • `PrivateSubnet2` — ID della seconda sottorete privata • `SqlFSxFCIName` — Nome del tag applicato ai nodi primari e secondari e ad Amazon FSx.	

Operazione	Description	Competenze richieste
	• `SqlFSxServerNetBIOSName1` — Nome del nodo dell' EC2 istanza principale (massimo 15 caratteri) • `SqlFSxServerNetBIOSName2` — Nome del nodo dell' EC2 istanza secondaria (massimo 15 caratteri) • `VPC`— ID VPC • `WorkloadInstanceType` — Tipo di istanza EC2 Distribuisci lo `infra` stack. Lo stack creerà tutti i componenti dell'infrastruttura necessari per configurare Windows SQL Server FCI. 2. Dopo l'avvio dei nodi dell' EC2 istanza, verrà richiamato il documento Systems Manager Automation per unire queste istanze ad AWS Managed Microsoft AD. È possibile tenere traccia dell'avanzamento nella pagina di automazione della console Systems Manager.	

Configura Windows SQL Server Always On FCI

Operazione	Description	Competenze richieste

Installa gli strumenti di Windows.	1. Accedi all' EC2 istanza principale, che è il nodo 1. Per installare le funzionalità di Windows (Active Directory e FCI Tools), esegui PowerShell lo script seguente.
	`Install-WindowsFeature -Name RSAT-AD-Powershell,Failover-Clustering -IncludeManagementTools`
	`Install-WindowsFeature -Name RSAT-Clustering,RSAT-ADDS-Tools,RSAT-AD-Powershell,RSAT-DHCP,RSAT-DNS-Server`
	2. Accedi all' EC2 istanza secondaria, che è il nodo 2, ed esegui lo stesso script per abilitare le funzionalità sul nodo 2.
		AWS DevOps, DevOps ingegnere, DBA

Preimposta gli oggetti del computer del cluster in Active Directory Domain Services.	Per preimpostare il cluster name object (CNO) in Active Directory Domain Services (AD DS) e preimpostare un oggetto computer virtuale (VCO) per un ruolo in cluster, segui le istruzioni nella documentazione di Windows Server.	AWS DevOps, DBA, DevOps ingegnere

Operazione	Description	Competenze richieste
Crea il WSFC.	Per creare il cluster Windows Server Failover Clustering (WSFC), procedi come segue: 1. Accedere all' EC2 istanza principale, che è il nodo 1. Per creare la condivisione di FSx file Amazon e concedere l'accesso completo all'account del servizio AD elencato, esegui il codice seguente. `Invoke-Command -ComputerName "<FSx Windows Remote PowerShell Endpoint>" -ConfigurationName FSxRemoteAdmin -scriptblock { New-FSxSmbShare -Name "SQLDB" -Path "D:\share" -Description "SQL Databases Share" -ContinuouslyAvailable $true -FolderEnumerationMode AccessBased -EncryptData $true grant-fsxsmb shareaccess -name SQLDB -AccountName "<domain\user>" -accessRight Full }` Questo comando creerà anche la condivisione di	AWS DevOps, DBA, DevOps ingegnere

Operazione	Description	Competenze richieste
	file a disponibilità continua (CA), ottimizzata per l'uso da parte di Microsoft SQL Server. 2. Per creare il cluster di failover sull'istanza principale (nodo 1), esegui il comando seguente. `New-Cluster -Name <CNO Name> -Node <Node1 Name>, <Node2 Name> -StaticAddress <Node1 Secondary Private IP>, <Node2 Secondary Private IP>` Il comando richiede i seguenti parametri: • `Name`— Il nome del cluster (CNO) • `Node`— I nomi dei nodi primari e secondari, rispettivamente • `StaticAddress` — Gli indirizzi IP secondari dei nodi primario e secondario, rispettivamente **Important** Un amministratore di dominio o un utente normale deve disporre	

Operazione	Description	Competenze richieste
	dell'autorizzazione e di amministratore su entrambi i nodi per creare il cluster Windows Server Failover Clustering (WSFC). In caso contrario, il comando precedente avrà esito negativo e restituirà il messaggio,. `You do not have administrator privilege on servers` 3. Dopo aver creato il cluster, esegui il comando seguente per allegare il file share witness. `Set-ClusterQuorum -FileShareWitness \ \<FSx Windows Remote PowerShell Endpoint> \share\witness`	

Operazione	Description	Competenze richieste
Installa il cluster di failover di SQL Server.	Dopo aver configurato il cluster WSFC, installa il cluster SQL Server sull'istanza principale (node1). 1. Nell'unità T su entrambi i nodi, crea e cartelle. `tempdb` **`log`** Le cartelle vengono utilizzate nei PowerShell comandi. 2. Dopo aver copiato i file multimediali di SQL Server per l'installazione di SQL Server su entrambi i nodi, esegui il PowerShell comando seguente sul nodo 1 per installare SQL Server sul nodo 1. `D:\setup.exe /Q ` `/ACTION=InstallFailoverCluster ` `/IACCEPTSQLSERVERLICENSETERMS ` `/FEATURES="SQL,IS,BC,Conn" ` `/INSTALLSHAREDDIR="C:\Program Files\Microsoft SQL Server" ` `/INSTALLSHAREDWOWDIR="C:\Program Files (x86)\Microsoft SQL Server" ` `/RSINSTALLMODE="FilesOnlyMode" ` `/INSTANCEID="MSSQLSERVER" `	AWS DevOps, DBA, DevOps ingegnere

Operazione	Description	Competenze richieste
	/INSTANCENAME="MSSQLSERVER" ` /FAILOVERCLUSTERGROUP="SQL Server (MSSQLSERVER)" ` /FAILOVERCLUSTERIPADDRESSES="IPv4;<2nd Sec Private Ip node1>;Cluster Network 1;<subnet mask>" ` /FAILOVERCLUSTERNETWORKNAME="<Fail over cluster Network Name>" ` /INSTANCEDIR="C:\Program Files\Microsoft SQL Server" ` /ENU="True" ` /ERRORREPORTING=0 ` /SQMREPORTING=0 ` /SAPWD="<Domain User password>" ` /SQLCOLLATION="SQL_Latin1_General_CP1_CI_AS" ` /SQLSYSADMINACCOUNTS="<domain\username>" ` /SQLSVCACCOUNT="<domain\username>" /SQLSVCPASSWORD="<Domain User password>" ` /AGTSVCACCOUNT="<domain\username>" /AGTSVCPASSWORD="<Domain User password>" ` /ISSVCACCOUNT="<domain\username>" /ISSVCPASSWORD="<Domain User password>" `	

Operazione	Description	Competenze richieste
	`/FTSVCACCOUNT="NT Service\MSSQLFDLauncher" ` `/INSTALLSQLDATADIR="\\<FSX DNS name>\share\Program Files\Microsoft SQL Server" ` `/SQLUSERDBDIR="\\<FSX DNS name>\share\data" ` `/SQLUSERDBLOGDIR="\\<FSX DNS name>\share\log" ` `/SQLTEMPDBDIR="T:\tempdb" ` `/SQLTEMPDBLOGDIR="T:\log" ` `/SQLBACKUPDIR="\\<FSX DNS name>\share\SQLBackup" ` `/SkipRules=Cluster_VerifyForErrors ` `/INDICATEPROGRESS`	

Operazione	Description	Competenze richieste
Aggiungi un nodo secondario al cluster.	Per aggiungere SQL Server al nodo secondario (nodo 2), esegui il PowerShell comando seguente. `D:\setup.exe /Q ` `/ACTION=AddNode ` `/IACCEPTSQLSERVERLICENSETERMS ` `/INSTANCENAME="MSSQLSERVER" ` `/FAILOVERCLUSTERGROUP="SQL Server (MSSQLSERVER)" ` `/FAILOVERCLUSTERIPADDRESSES="IPv4;<2nd Sec Private Ip node2>;Cluster Network 2;<subnet mask>" ` `/FAILOVERCLUSTERNETWORKNAME="<Fail over cluster Network Name>" ` `/CONFIRMIPDEPENDENCYCHANGE=1 ` `/SQLSVCACCOUNT="<domain\username>" /SQLSVCPASSWORD="<Domain User password>" ` `/AGTSVCACCOUNT="domain\username>" /AGTSVCPASSWORD="<Domain User password>" ` `/FTSVCACCOUNT="NT Service\MSSQLFDLauncher" ` `/SkipRules=Cluster_VerifyForErrors `	AWS DevOps, DBA, DevOps ingegnere

Operazione	Description	Competenze richieste
	`/INDICATEPROGRESS`	
Prova la FCI di SQL Server.	1. Nell'istanza Windows di uno dei nodi, in Strumenti di amministrazione, avvia Failover Cluster Manager. 2. Passa a Nodes e conferma che lo stato del nodo sia Status Running. 3. Seleziona Ruoli, apri il menu contestuale (con il pulsante destro del mouse) per SQL Server (MSSQLSERVER) e seleziona Sposta e seleziona nodo. 4. Dopo la selezione del nodo, SQL Server dovrebbe essere in esecuzione sull'altro nodo.	DBA, ingegnere DevOps

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Eliminare le risorse.	Per ripulire le risorse, utilizza il processo di eliminazione CloudFormation dello stack AWS: 1. Apri la CloudFormation console AWS. 2. Nella pagina Stacks, seleziona lo `infra` stack.	AWS DevOps, DBA, DevOps ingegnere

Operazione	Description	Competenze richieste
	Lo stack deve essere attualmente in esecuzione. 3. Nel riquadro dei dettagli dello stack, scegliere Delete (Elimina). 4. Selezionare Delete stack (Elimina stack) quando richiesto. 5. Ripeti i passaggi 2-4 per lo stack. `ssm` Una volta completata l'elimina zione dello stack, gli stack saranno nello stato in cui si trova. `DELETE_COMPLETE` Per impostazione predefini ta, gli stack nello `DELETE_CO MPLETE` stato non vengono visualizzati nella CloudForm ation console. Per visualizz are gli stack eliminati, devi modificare il filtro di visualizz azione dello stack come descritto in Visualizzazione degli stack eliminati sulla console AWS. CloudFormation Se l'eliminazione non è riuscita, uno stack sarà nello stato. `DELETE_FAILED` Per le soluzioni, consulta Delete stack fail nella CloudFormation documentazione.	

risoluzione dei problemi

Problema	Soluzione
Errore CloudFormation del modello AWS	Se il CloudFormation modello fallisce durante la distribuzione, procedi come segue: 1. Apri la CloudFormation console AWS. 2. Nella pagina Stacks della CloudFormation console, seleziona lo stack. 3. Scegli Eventi e controlla lo stato dello stack.
Connessione AWS Managed Microsoft AD non riuscita	Per risolvere i problemi di iscrizione, segui questi passaggi: 1. Aprire la console Systems Manager. 2. Seleziona la regione di distribuzione. 3. Nel riquadro di sinistra, scegli Automazione e individua il runbook di automazione non riuscito. 4. Apri il runbook di automazione e verifica lo stato di esecuzione e i passaggi di esecuzione. 5. Esamina i dettagli del passaggio non riuscito per vedere l'errore o l'errore esatto.

Risorse correlate

- Semplifica le distribuzioni ad alta disponibilità di Microsoft SQL Server utilizzando Amazon FSx per Windows File Server
- Utilizzo FSx per Windows File Server con Microsoft SQL Server

Usa le query BMC Discovery per estrarre i dati di migrazione per la pianificazione della migrazione

Ben Tailor-Hamblin, Emma Baldry, Simon Cunningham e Shabnam Khan, Amazon Web Services

Riepilogo

Questa guida fornisce esempi di query e procedure per aiutarvi a estrarre dati dall'infrastruttura e dalle applicazioni locali utilizzando BMC Discovery. Lo schema mostra come utilizzare le query BMC Discovery per scansionare l'infrastruttura ed estrarre informazioni su software, servizi e dipendenze. I dati estratti sono necessari per le fasi di valutazione e mobilitazione di una migrazione su larga scala verso il cloud Amazon Web Services (AWS). Puoi utilizzare questi dati per prendere decisioni critiche su quali applicazioni migrare insieme come parte del tuo piano di migrazione.

Prerequisiti e limitazioni

Prerequisiti

- Una licenza per BMC Discovery (precedentemente BMC ADDM) o la versione software as a service (SaaS) di BMC Helix Discovery
- Versione locale o SaaS di BMC Discovery, installata

> **Note**
>
> Per le versioni locali di BMC Discovery, è necessario installare l'applicazione su una rete client con accesso a tutti i dispositivi di rete e server che sono oggetto di una migrazione tra più data center. L'accesso alla rete client deve essere fornito in base alle istruzioni di installazione dell'applicazione. Se è richiesta la scansione delle informazioni di Windows Server, è necessario configurare un dispositivo di gestione proxy Windows nella rete.

- Accesso alla rete per consentire all'applicazione di scansionare i dispositivi tra i data center, se si utilizza BMC Helix Discovery

Versioni del prodotto

- BMC Discovery 22.2 (12.5)
- BMC Discovery 22.1 (12.4)
- BMC Discovery 21.3 (12.3)

- BMC Discovery 21.05 (12.2)
- BMC Discovery 20.08 (12.1)
- BMC Discovery 20.02 (12.0)
- BMC Discovery 11.3
- BMC Discovery 11.2
- BMC Discovery 11.1
- BMC Discovery 11.0
- BMC Atrium Discovery 10.2
- BMC Atrium Discovery 10.1
- BMC Atrium Discovery 10.0

Architecture

Il diagramma seguente mostra come gli asset manager possono utilizzare le query BMC Discovery per scansionare applicazioni modellate BMC in ambienti SaaS e locali.

Il diagramma mostra il seguente flusso di lavoro: Un asset manager utilizza BMC Discovery o BMC Helix Discovery per scansionare le istanze di database e software in esecuzione su server virtuali ospitati su più server fisici. Lo strumento può modellare applicazioni con componenti che si estendono su più server virtuali e fisici.

Stack tecnologico

- BMC Discovery
- BMC Helix Discovery

Tools (Strumenti)

- BMC Discovery è uno strumento di rilevamento dei data center che consente di individuare automaticamente il data center.
- BMC Helix Discovery è un sistema di rilevamento e modellazione delle dipendenze basato su SaaS che consente di modellare dinamicamente gli asset di dati e le relative dipendenze.

Best practice

È consigliabile mappare i dati delle applicazioni, delle dipendenze e dell'infrastruttura durante la migrazione al cloud. La mappatura consente di comprendere la complessità dell'ambiente corrente e le dipendenze tra i vari componenti.

Le informazioni sugli asset fornite da queste interrogazioni sono importanti per diversi motivi:

1. Pianificazione: la comprensione delle dipendenze tra i componenti consente di pianificare il processo di migrazione in modo più efficace. Ad esempio, potrebbe essere necessario migrare prima alcuni componenti per garantire che altri possano essere migrati correttamente.
2. Valutazione dei rischi: la mappatura delle dipendenze tra i componenti può aiutarti a identificare eventuali rischi o problemi potenziali che possono sorgere durante il processo di migrazione. Ad esempio, potresti scoprire che alcuni componenti si basano su tecnologie obsolete o non supportate che potrebbero causare problemi nel cloud.
3. Architettura cloud: la mappatura dei dati delle applicazioni e dell'infrastruttura può anche aiutarti a progettare un'architettura cloud adatta che soddisfi le tue esigenze organizzative. Ad esempio, potrebbe essere necessario progettare un'architettura a più livelli per supportare requisiti di alta disponibilità o scalabilità.

Nel complesso, la mappatura dei dati delle applicazioni, delle dipendenze e dell'infrastruttura è un passaggio cruciale nel processo di migrazione al cloud. L'esercizio di mappatura può aiutarti a comprendere meglio il tuo ambiente attuale, identificare eventuali problemi o rischi potenziali e progettare un'architettura cloud adeguata.

Epiche

Identifica e valuta gli strumenti di scoperta

Operazione	Description	Competenze richieste
Identifica i proprietari di ITSM.	Identifica i proprietari dell'IT Service Management (ITSM) (di solito contattando i team di supporto operativo).	Responsabile della migrazione
Controlla CMDB.	Identifica il numero di database di gestione della	Responsabile della migrazione

Operazione	Description	Competenze richieste
	configurazione (CMDBs) che contengono informazioni sugli asset, quindi identifica le fonti di tali informazioni.	
Identifica gli strumenti di scoperta e verifica l'utilizzo di BMC Discovery.	Se la tua organizzazione utilizza BMC Discovery per inviare dati sull'ambiente allo strumento CMDB, verifica l'ambito e la copertura delle sue scansioni. Ad esempio, controlla se BMC Discovery sta scansionando tutti i data center e se i server di accesso si trovano in zone perimetrali.	Responsabile della migrazione
Verifica il livello di modellazione delle applicazioni.	Verifica se le applicazioni sono modellate in BMC Discovery. In caso contrario, consigliamo di utilizzare lo strumento BMC Discovery per modellare quali istanze software in esecuzione e forniscono un'applicazione e un servizio aziendale.	Ingegnere addetto alla migrazione, responsabile della migrazione

Estrarre i dati dell'infrastruttura

Operazione	Description	Competenze richieste
Estrai dati su server fisici e virtuali.	Per estrarre dati dai server fisici e virtuali scansionati da BMC Discovery, utilizzare Query Builder per eseguire la seguente query:	Ingegnere addetto alla migrazione, responsabile della migrazione

Operazione	Description	Competenze richieste
	`search Host show key as 'Serverid', virtual, name as 'HOSTNAME', os_type as 'osName', os_version as 'OS Version', num_logical_processors as 'Logical Processor Counts', cores_per_processor as 'Cores per Processor', logical_ram as 'Logical RAM', #Consumer:StorageUse:Provider:DiskDrive.size as 'Size'` **Note** È possibile utilizzare i dati estratti per determinare le dimensioni delle istanze appropriate per la migrazione.	

Operazione	Description	Competenze richieste
Estrai dati su applicazioni modellate.	Se le applicazioni sono modellate in BMC Discovery , è possibile estrarre dati sui server che eseguono il software applicativo. Per ottenere i nomi dei server, utilizzare Query Builder per eseguire la seguente query: `search SoftwareInstance show key as 'ApplicationID', #RunningSoftware:HostedSoftware:Host:Host.key as 'ReferenceID', type, name` **Note** Le applicazioni sono modellate in BMC Discovery mediante una raccolta di istanze software in esecuzione. L'applicazione dipende da tutti i server che eseguono il software applicativo.	Proprietario dell'applicazione BMC Discovery

Operazione	Description	Competenze richieste
Estrarre dati dai database.	Per ottenere un elenco di tutti i database analizzati e dei server su cui sono in esecuzione questi database, utilizzate Query Builder per eseguire la seguente query: `search Database show key as 'Key', name, type as 'Source Engine Type', #Detail:Detail:ElementWithDetail:SoftwareInstance.name as 'Software Instance', #Detail:Detail:ElementWithDetail:SoftwareInstance.product_version as 'Product Version', #Detail:Detail:ElementWithDetail:SoftwareInstance.edition as 'Edition', #Detail:Detail:ElementWithDetail:SoftwareInstance.#RunningSoftware:HostedSoftware:Host:Host.key as 'ServerID'`	Proprietario dell'app

Operazione	Description	Competenze richieste
Estrai dati sulla comunicazione con il server.	Per ottenere informazioni su tutte le comunicazioni di rete tra i server raccolte da BMC Discovery dai registri storici delle comunicazioni di rete, utilizzare Query Builder per eseguire la seguente query: `search Host TRAVERSE InferredElement:Inference:Associate:DiscoveryAccess TRAVERSE DiscoveryAccess:DiscoveryAccessResult:DiscoveryResult:NetworkConnectionList TRAVERSE List:List:Member:DiscoveredNetworkConnection PROCESS WITH networkConnectionInfo`	Proprietario dell'applicazione BMC Discovery
Estrai i dati sulla scoperta delle applicazioni.	Per ottenere informazioni sulle dipendenze delle applicazioni, utilizzate Query Builder per eseguire la seguente query: `search SoftwareInstance show key as 'SRC App ID', #Dependant:Dependency:DependedUpon:SoftwareInstance.key as 'DEST App ID'`	Proprietario dell'applicazione BMC Discovery

Operazione	Description	Competenze richieste
Estrai dati sui servizi aziendali.	Per estrarre dati sui servizi aziendali forniti dagli host, utilizza Query Builder per eseguire la seguente query: `search Host show name, #Host:HostedSoftware:AggregateSoftware:BusinessService.name as 'Name'`	Proprietario dell'applicazione BMC Discovery

risoluzione dei problemi

Problema	Soluzione
Una query non viene eseguita o contiene colonne non popolate.	Esamina i record degli asset in BMC Discovery e determina quali campi sono necessari. Quindi, sostituite questi campi nella query utilizzando Query Builder.
I dettagli di una risorsa dipendente non vengono compilati.	Ciò è probabilmente dovuto alle autorizzazioni di accesso o alla connettività di rete. Lo strumento di rilevamento potrebbe non disporre delle autorizzazioni necessarie per accedere a determinate risorse, in particolare se si trovano su reti o ambienti diversi. Ti consigliamo di lavorare a stretto contatto con esperti in materia di discovery per garantire che tutte le risorse pertinenti vengano identificate.

Risorse correlate

Riferimenti

- Diritto alla licenza BMC Discovery (documentazione BMC)
- Caratteristiche e componenti di BMC Discovery (documentazione BMC)
- Guida per l'utente di BMC Discovery (documentazione BMC)
- Ricerca di dati (su BMC Discovery) (documentazione BMC)
- Individuazione e analisi del portafoglio per la migrazione (AWS Prescriptive Guidance)

Tutorial e video

- BMC Discovery: Webinar - Le migliori pratiche per la creazione di report sulle interrogazioni (parte 1) () YouTube

Trasferisci

Argomenti

- Migrazione di un database Amazon RDS for Oracle su Account AWS un altro database Regione AWS e AWS DMS utilizzo per la replica continua
- Esegui la migrazione di un'istanza database Amazon RDS su un altro VPC o account
- Esegui la migrazione di un cluster Amazon Redshift in una regione AWS in Cina
- Trasporta i database PostgreSQL tra due istanze DB Amazon RDS utilizzando pg_transport

Migrazione di un database Amazon RDS for Oracle su Account AWS un altro database Regione AWS e AWS DMS utilizzo per la replica continua

Durga Prasad Cheepuri e Eduardo Valentim, Amazon Web Services

Riepilogo

Attenzione: gli utenti IAM dispongono di credenziali a lungo termine, il che rappresenta un rischio per la sicurezza. Per ridurre questo rischio, si consiglia di fornire a questi utenti solo le autorizza zioni necessarie per eseguire l'attività e di rimuoverli quando non sono più necessari.

Questo modello illustra i passaggi per la migrazione di un database di origine Amazon Relational Database Service (Amazon RDS) per Oracle su un altro database e. Account AWS Regione AWS Il modello utilizza uno snapshot DB per un caricamento completo dei dati una tantum e abilita AWS Database Migration Service ()AWS DMS la replica continua.

Prerequisiti e limitazioni

Prerequisiti

- Un file attivo Account AWS che contiene il database Amazon RDS for Oracle di origine, che è stato crittografato utilizzando una chiave AWS Key Management Service non predefinita AWS KMS()
- Attivo Account AWS in un database diverso Regione AWS da quello di origine, da utilizzare per il database Amazon RDS for Oracle di destinazione
- Peering su cloud privato virtuale (VPC) tra l'origine e la destinazione VPCs
- Familiarità con l'utilizzo di un database Oracle come fonte per AWS DMS
- Familiarità con l'utilizzo di un database Oracle come destinazione per AWS DMS

Versioni del prodotto

- Versioni Oracle 11g (versioni 11.2.0.3.v1 e successive) e fino a 12.2 e 18c. Per l'elenco più recente delle versioni ed edizioni supportate, vedere Utilizzo di un database Oracle come origine per AWS DMS e con utilizzo di un database Oracle come destinazione nella documentazione. AWS DMS AWS Per le versioni Oracle supportate da Amazon RDS, consulta Oracle su Amazon RDS.

Architecture

Stack tecnologici di origine e destinazione

- Istanza database Amazon RDS per Oracle

Architettura di replica continua

Tools (Strumenti)

Strumenti utilizzati per il caricamento completo dei dati una tantum

- Amazon Relational Database Service (Amazon RDS) crea uno snapshot del volume di storage dell'istanza DB, eseguendo il backup dell'intera istanza DB e non solo dei singoli database. Quando crei uno snapshot DB è necessario identificare qual è l'istanza database di cui stai effettuando il backup e dare un nome allo snapshot DB in modo da poterlo usare successivamente per il ripristino. La quantità di tempo necessaria per creare uno snapshot varia a seconda della dimensione dei database. Poiché lo snapshot include l'intero volume d'archiviazione, la dimensione dei file, come i file temporanei, influisce sulla quantità di tempo necessaria per creare lo snapshot. Per ulteriori informazioni sull'uso degli snapshot DB, consulta Creazione di uno snapshot DB nella documentazione di Amazon RDS.
- AWS Key Management Service (AWS KMS) crea una chiave per la crittografia Amazon RDS. Quando crei un'istanza DB crittografata, puoi anche fornire l'identificatore della AWS KMSchiave per la tua chiave di crittografia. Se non specifichi un identificatore di AWS KMSchiave, Amazon RDS utilizza la chiave di crittografia predefinita per la tua nuova istanza DB. AWS KMScrea la tua chiave di crittografia predefinita per il tuo. Account AWS La tua Account AWS ha una chiave di crittografia predefinita diversa per ciascuno Regione AWS. Per questo modello, l'istanza database di Amazon RDS deve essere crittografata utilizzando una chiave non predefinita AWS KMS. Per ulteriori informazioni sull'uso delle AWS KMSchiavi per la crittografia Amazon RDS, consulta Encrypting Amazon RDS resources nella documentazione di Amazon RDS.

Strumenti utilizzati per la replica continua

- AWS Database Migration Service (AWS DMS) viene utilizzato per replicare le modifiche in corso e per mantenere sincronizzati i database di origine e di destinazione. Per ulteriori informazioni

sull'utilizzo AWS DMS per la replica continua, consulta Lavorare con un'istanza di AWS DMS replica nella documentazione. AWS DMS

Epiche

Configura la tua fonte Account AWS

Operazione	Description	Competenze richieste
Preparare l'istanza database Oracle di origine.	Lascia che l'istanza DB di Amazon RDS for Oracle venga eseguita in modalità ARCHIVELOG e imposta il periodo di conservazione. Per i dettagli, consulta Lavorare con un database Oracle AWS gestito come fonte per. AWS DMS	DBA
Imposta la registrazione supplementare per l'istanza database Oracle di origine.	Imposta la registrazione supplementare a livello di database e tabella per l'istanza DB Amazon RDS for Oracle. Per i dettagli, consulta Lavorare con un AWS database Oracle gestito come fonte per. AWS DMS	DBA
Aggiorna la politica AWS KMS chiave nell'account di origine.	Aggiorna la policy delle AWS KMS chiavi nell'origine Account AWS per consentir e alla destinazione Account AWS di utilizzare la AWS KMS chiave Amazon RDS crittogra fata. Per i dettagli, consulta la AWS KMS documentazione.	SysAdmin

Operazione	Description	Competenze richieste
Crea uno snapshot Amazon RDS DB manuale dell'istanza DB di origine.		Utente AWS IAM
Condividi lo snapshot Amazon RDS manuale e crittografato con la destinazione. Account AWS	Per i dettagli, consulta Condivisione di uno snapshot DB.	Utente AWS IAM

Configura il tuo obiettivo Account AWS

Operazione	Description	Competenze richieste
Allega una politica.	Nella destinazione Account AWS, allega una policy AWS Identity and Access Management (IAM) all'utente IAM root, per consentire all'utente IAM di copiare uno snapshot DB crittografato utilizzando la AWS KMS chiave condivisa.	SysAdmin
Passa alla fonte Regione AWS.		Utente AWS IAM
Copia l'istantanea condivisa.	Nella console Amazon RDS, nel riquadro Snapshot, scegli Shared with Me e seleziona lo snapshot condiviso. Copia lo snapshot nello Regione AWS stesso database di origine utilizzando Amazon Resource Name (ARN) per AWS KMS la chiave utilizzata	Utente AWS IAM

Operazione	Description	Competenze richieste
	dal database di origine. Per i dettagli, consulta Copiare uno snapshot DB.	
Passa alla destinazione Regione AWS e crea una nuova AWS KMS chiave.		Utente AWS IAM
Copia l'istantanea.	Passa alla fonte Regione AWS. Sulla console Amazon RDS, nel riquadro Snapshot, scegli Owned by Me e seleziona lo snapshot copiato. Copia lo snapshot sulla destinazione Regione AWS utilizzando la AWS KMS chiave per la nuova destinazi one. Regione AWS	Utente AWS IAM
Ripristinare lo snapshot:	Passa al bersaglio Regione AWS. Sulla console Amazon RDS, nel riquadro Snapshot, scegli Owned by Me. Seleziona lo snapshot copiato e ripristinalo su un'istanza DB Amazon RDS for Oracle. Per i dettagli, consulta Ripristino da uno snapshot DB.	Utente AWS IAM

Prepara il tuo database di origine per la replica continua

Operazione	Description	Competenze richieste
Crea un utente Oracle con le autorizzazioni appropriate.	Crea un utente Oracle con i privilegi richiesti per Oracle	DBA

Operazione	Description	Competenze richieste
	come fonte per. AWS DMS Per i dettagli, consulta la AWS DMS documentazione.	
Configura il database di origine per Oracle LogMiner o Oracle Binary Reader.		DBA

Prepara il database di destinazione per la replica continua

Operazione	Description	Competenze richieste
Crea un utente Oracle con le autorizzazioni appropriate.	Crea un utente Oracle con i privilegi richiesti per Oracle come destinazione per. AWS DMS Per i dettagli, consulta la AWS DMS documentazione.	DBA

Crea componenti AWS DMS

Operazione	Description	Competenze richieste
Crea un'istanza di replica nella destinazione Regione AWS.	Crea un'istanza di replica nel VPC della destinazione. Regione AWSPer i dettagli, consulta la documenta zione.AWS DMS	Utente AWS IAM
Crea endpoint di origine e destinazione con la crittografia richiesta e testa le connessio ni.	Per i dettagli, consulta la AWS DMS documentazione.	DBA
Crea attività di replica.	1. Per il tipo di migrazione, scegli la replica continua.	Utente IAM

Operazione	Description	Competenze richieste
	2. Per il punto di partenza dell'acquisizione dei dati di modifica (CDC), utilizza il numero di modifica del sistema Oracle (SCN) quando lo snapshot di Amazon RDS è stato acquisito a pieno carico o il timestamp quando è stato eseguito il caricamento completo. 3. Per **`TargetTablePrepMode`**, scegli DO_NOTHING. Se l'attività ha tabelle di dati LOB (Large Binary Object), scegliete la modalità LOB limitata e impostate la dimensione massima del LOB sulla dimensione massima dei dati LOB nella tabella. 4. Attivare la registrazione nel log. 5. Raggruppa le tabelle correlate tramite chiavi in un'unica attività. Se sono presenti tabelle con una grande quantità di dati LOB e la tabella non ha alcuna relazione con altre tabelle, create un'attività separata con le impostazioni LOB descritte in precedenza.	

Operazione	Description	Competenze richieste
	Per i dettagli, consultate la documentazione.AWS DMS	
Avvia le attività e monitorale.	Per i dettagli, consulta la AWS DMS documentazione.	Utente AWS IAM
Abilita la convalida dell'attività, se necessario.	Si noti che l'abilitazione della convalida ha un impatto sulle prestazioni della replica. Per i dettagli, consulta la documentazione.AWS DMS	Utente AWS IAM

Risorse correlate

- Modifica di una politica chiave
- Creazione di uno snapshot Amazon RDS DB manuale
- Condivisione di uno snapshot Amazon RDS DB manuale
- Copiare uno snapshot
- Ripristino da uno snapshot Amazon RDS DB
- Iniziare con AWS DMS
- Utilizzo di un database Oracle come fonte per AWS DMS
- Utilizzo di un database Oracle come destinazione per AWS DMS
- AWS DMS configurazione tramite peering VPC
- Come posso condividere istantanee manuali di Amazon RDS DB o snapshot di cluster DB con un altro? Account AWS (articolo dell'AWS Knowledge Center)

Esegui la migrazione di un'istanza database Amazon RDS su un altro VPC o account

Dhrubajyoti Mukherjee, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni per la migrazione di un'istanza DB Amazon Relational Database Service (Amazon RDS) da un cloud privato virtuale (VPC) a un altro nello stesso account AWS o da un account AWS a un altro account AWS.

Questo modello è utile se desideri migrare le istanze database di Amazon RDS su un altro VPC o account per motivi di separazione o sicurezza (ad esempio, quando desideri posizionare lo stack di applicazioni e il database in modo diverso). VPCs

La migrazione di un'istanza DB verso un altro account AWS prevede passaggi come l'acquisizione di uno snapshot manuale, la condivisione e il ripristino dello snapshot nell'account di destinazione. Questo processo può richiedere molto tempo, a seconda delle modifiche al database e dei tassi di transazione. Inoltre, causa tempi di inattività del database, quindi pianifica in anticipo la migrazione. Prendi in considerazione una strategia di blue/green implementazione per ridurre al minimo i tempi di inattività. In alternativa, puoi valutare AWS Data Migration Service (AWS DMS) per ridurre al minimo i tempi di inattività dovuti alla modifica. Tuttavia, questo modello non copre questa opzione. Per ulteriori informazioni, consulta la documentazione di AWS DMS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Autorizzazioni AWS Identity and Access Management (IAM) richieste per VPC, sottoreti e console Amazon RDS

Limitazioni

- Le modifiche a un VPC provocano il riavvio del database, con conseguenti interruzioni delle applicazioni. Si consiglia di eseguire la migrazione durante i periodi di picco più bassi.
- Limitazioni durante la migrazione di Amazon RDS su un altro VPC:
  - L'istanza DB da migrare deve essere una singola istanza senza standby. Non deve essere membro di un cluster.
  - Amazon RDS non deve trovarsi in più zone di disponibilità.

  - Amazon RDS non deve avere alcuna replica di lettura.
  - Il gruppo di sottoreti creato nel VPC di destinazione deve avere sottoreti della zona di disponibilità in cui è in esecuzione il database di origine.
- Limitazioni durante la migrazione di Amazon RDS a un altro account AWS:
  - La condivisione di istantanee crittografate con la chiave di servizio predefinita per Amazon RDS non è attualmente supportata.

Architecture

Migrazione a un VPC nello stesso account AWS

Il diagramma seguente mostra il flusso di lavoro per la migrazione di un'istanza DB Amazon RDS su un VPC diverso nello stesso account AWS.

I passaggi sono i seguenti. Consulta la sezione Epic per istruzioni dettagliate.

1. Crea un gruppo di sottoreti DB nel VPC di destinazione. Un gruppo di sottoreti DB è una raccolta di sottoreti che è possibile utilizzare per specificare un VPC specifico quando si creano istanze DB.
2. Configura l'istanza DB di Amazon RDS nel VPC di origine per utilizzare il nuovo gruppo di sottoreti DB.
3. Applica le modifiche per migrare il database Amazon RDS al VPC di destinazione.

Migrazione a un altro account AWS

Il diagramma seguente mostra il flusso di lavoro per la migrazione di un'istanza DB Amazon RDS su un altro account AWS.

I passaggi sono costituiti dai seguenti. Consulta la sezione Epic per istruzioni dettagliate.

1. Accedi all'istanza database Amazon RDS nell'account AWS di origine.
2. Crea uno snapshot Amazon RDS nell'account AWS di origine.
3. Condividi lo snapshot Amazon RDS con l'account AWS di destinazione.
4. Accedi allo snapshot Amazon RDS nell'account AWS di destinazione.
5. Crea un'istanza database Amazon RDS nell'account AWS di destinazione.

Tools (Strumenti)

Servizi AWS

- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito. Questa rete virtuale è simile a una rete tradizionale che gestiresti nel tuo data center, con i vantaggi dell'utilizzo dell'infrastruttura scalabile di AWS.

Best practice

- Se il downtime del database è un problema durante la migrazione di un'istanza DB Amazon RDS su un altro account, ti consigliamo di utilizzare AWS DMS. Questo servizio fornisce la replica dei dati, che causa meno di cinque minuti di interruzione.

Epiche

Esegui la migrazione a un altro VPC nello stesso account AWS

Operazione	Description	Competenze richieste
Crea un nuovo VPC.	Sulla console Amazon VPC, crea un nuovo VPC e sottoreti con le proprietà e gli intervalli di indirizzi IP desiderati. Per istruzioni dettagliate, consulta la documentazione di Amazon VPC.	Amministratore
Creazione di un gruppo di sottoreti del database.	Sulla console Amazon RDS: 1. Scegli Gruppi di sottoreti, Crea gruppo di sottoreti DB. 2. Immettere il nome, la descrizione e l'ID VPC del gruppo di sottoreti.	Amministratore

Operazione	Description	Competenze richieste
	3. Aggiungi le sottoreti che appartengono al gruppo di sottoreti. Aggiungi sottoreti per coprire almeno due zone di disponibilità. 4. Scegli Create (Crea). Per ulteriori informazioni, consulta la documentazione di Amazon RDS.	

Operazione	Description	Competenze richieste
Modifica l'istanza DB di Amazon RDS per utilizzare il nuovo gruppo di sottoreti.	Sulla console Amazon RDS: 1. Nel riquadro di navigazione, scegli Database, quindi scegli l'istanza database di Amazon RDS da migrare. 2. Nella sezione Connettività, scegli il gruppo di sottoreti associato al VPC di destinazione. 3. Nella sezione Modifiche alla pianificazione, scegli Applica immediatamente. Una volta completata la migrazione al VPC di destinazione, il gruppo di sicurezza predefinito del VPC di destinazione viene assegnato all'istanza database di Amazon RDS. Puoi configurare un nuovo gruppo di sicurezza per quel VPC con le regole in entrata e in uscita richieste per la tua istanza DB. In alternativa, utilizza l'AWS Command Line Interface (AWS CLI) per eseguire la migrazione al VPC di destinazione fornendo esplicitamente il nuovo ID del gruppo di sicurezza VPC. Esempio:	Amministratore

Operazione	Description	Competenze richieste
	`aws rds modify-db-instance \ --db-instance-identifier testrds \ --db-subnet-group-name new-vpc-subnet-group \ --vpc-security-group-ids sg-idxxxx \ --apply-immediately`	

Esegui la migrazione a un altro account AWS

Operazione	Description	Competenze richieste
Crea un nuovo VPC e un nuovo gruppo di sottoreti nell'account AWS di destinazione.	1. Sulla console Amazon VPC, crea un nuovo VPC con le proprietà e gli intervalli di indirizzi IP desiderati. Per istruzioni dettagliate, consulta la documentazione di Amazon VPC. 2. Crea sottoreti per il nuovo VPC seguendo le istruzioni nella documentazione di Amazon VPC. 3. Sulla console Amazon RDS, crea gruppi di sottoreti DB. Per istruzioni, consulta la documentazione di Amazon RDS.	Amministratore
Condividi uno snapshot manuale del database e	1. Crea uno snapshot manuale del database	Amministratore

Operazione	Description	Competenze richieste
condividilo con l'account di destinazione.	di origine seguendo le istruzioni nella documenta zione di Amazon RDS. 2. Condividi lo snapshot con l'account AWS di destinazi one fornendo l'ID dell'acco unt di destinazione. Per istruzioni, consulta l'articolo di re:POST sulla condivisi one di snapshot DB con altri account.	
Avvia una nuova istanza database Amazon RDS.	Avvia una nuova istanza Amazon RDS DB dallo snapshot condiviso nell'acco unt AWS di destinazione. Per istruzioni, consulta la documentazione di Amazon RDS.	Amministratore

Risorse correlate

- Documentazione Amazon VPC
- Documentazione Amazon RDS
- Come posso modificare il VPC per un'istanza DB RDS? (Articolo AWS Re:Post)
- In che modo posso trasferire la proprietà delle risorse Amazon RDS a un altro account AWS? (Articolo AWS Re:Post)
- In che modo posso condividere snapshot manuali di Amazon RDS DB o snapshot di cluster Aurora DB con un altro account AWS? (Articolo AWS Re:Post)
- Documentazione AWS DMS

Esegui la migrazione di un cluster Amazon Redshift in una regione AWS in Cina

Jing Yan, Amazon Web Services

Riepilogo

Questo modello fornisce un step-by-step approccio per migrare un cluster Amazon Redshift in una regione AWS in Cina da un'altra regione AWS.

Questo modello utilizza i comandi SQL per ricreare tutti gli oggetti del database e utilizza il comando UNLOAD per spostare questi dati da Amazon Redshift a un bucket Amazon Simple Storage Service (Amazon S3) nella regione di origine. I dati vengono quindi migrati in un bucket S3 nella regione AWS in Cina. Il comando COPY viene utilizzato per caricare i dati dal bucket S3 e trasferirli al cluster Amazon Redshift di destinazione.

Amazon Redshift attualmente non supporta funzionalità interregionali come la copia di snapshot nelle regioni AWS in Cina. Questo modello fornisce un modo per aggirare tale limitazione. Puoi anche invertire i passaggi di questo schema per migrare i dati da una regione AWS in Cina a un'altra regione AWS.

Prerequisiti e limitazioni

Prerequisiti

- Account AWS attivi sia in una regione cinese che in una regione AWS al di fuori della Cina
- Cluster Amazon Redshift esistenti sia in una regione cinese che in una regione AWS al di fuori della Cina

Limitazioni

- Si tratta di una migrazione offline, il che significa che il cluster Amazon Redshift di origine non può eseguire operazioni di scrittura durante la migrazione.

Architecture

Stack tecnologico di origine

- Cluster Amazon Redshift in una regione AWS al di fuori della Cina

Stack tecnologico Target

- Cluster Amazon Redshift in una regione AWS in Cina

Architettura Target

Tools (Strumenti)

Strumenti

- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti che offre scalabilità, disponibilità dei dati, sicurezza e prestazioni. Puoi utilizzare Amazon S3 per archiviare dati da Amazon Redshift e copiare dati da un bucket S3 ad Amazon Redshift.
- Amazon Redshift — Amazon Redshift è un servizio di data warehouse completamente gestito su scala di petabyte nel cloud.
- psql — psql è un front-end basato su terminale per PostgreSQL.

Epiche

Preparati per la migrazione nella regione di origine

Operazione	Description	Competenze richieste
Avvia e configura un' EC2 istanza nella regione di origine.	Accedi alla Console di gestione AWS e apri la console Amazon Elastic Compute Cloud (Amazon EC2). La tua regione attuale viene visualizzata nella barra di navigazione nella parte superiore dello schermo. Questa regione non può essere una regione AWS in Cina. Dalla dashboard della EC2 console Amazon, scegli «Launch instance» e crea e configura un' EC2 istanza. Importante: assicurati che i	DBA, Sviluppatore

Operazione	Description	Competenze richieste
	gruppi EC2 di sicurezza per le regole in entrata consentano l'accesso illimitato alla porta TCP 22 dal tuo computer di origine. Per istruzioni su come avviare e configurare un' EC2 istanza, consulta la sezione «Risorse correlate».	
Installa lo strumento psql.	Scarica e installa PostgreSQL. Amazon Redshift non fornisce lo strumento psql, è installato con PostgreSQL. Per ulteriori informazioni sull'uso di psql e sull'installazione degli strumenti PostgreSQL, consulta la sezione «Risorse correlate».	DBA

Operazione	Description	Competenze richieste
Registra i dettagli del cluster Amazon Redshift.	Apri la console Amazon Redshift e scegli «Clusters » nel pannello di navigazio ne. Quindi scegli il nome del cluster Amazon Redshift dall'elenco. Nella scheda «Proprietà», nella sezione «Configurazioni del database» , registra il «Nome del database» e «Porta». <databasename>Apri la sezione «Dettagli di connessio ne» e registra l' "Endpoint», che è nel formato «endpoint :<port>/». Importante: assicurati che i gruppi di sicurezza di Amazon Redshift per le regole in entrata consentano l'accesso illimitato alla porta TCP 5439 dalla tua istanza. EC2	DBA
Connect psql al cluster Amazon Redshift.	<userid><databasen ame><port>Al prompt dei comandi, specifica le informazi oni di connessione eseguendo il comando «psql -h -U -d <endpoint>-p». <userid>A lla richiesta della password psql, inserisci la password per l'utente "». Verrai quindi connesso al cluster Amazon Redshift e potrai inserire comandi in modo interattivo.	DBA

Operazione	Description	Competenze richieste
Crea un bucket S3.	Apri la console Amazon S3 e crea un bucket S3 per contenere i file esportati da Amazon Redshift. Per istruzioni su come creare un bucket S3, consulta la sezione «Risorse correlate».	DBA, AWS Generale
Crea una policy IAM che supporti lo scaricamento dei dati.	Apri la console AWS Identity and Access Managemen t (IAM) e scegli «Policies ». Scegli «Crea policy» e scegli la scheda «JSON». Copia e incolla la politica IAM per lo scaricamento dei dati dalla sezione «Informaz ioni aggiuntive». Importante: sostituisci «s3_bucket_name» con il nome del tuo bucket S3. Scegli «Rivedi la politica» e inserisci un nome e una descrizione per la politica. Scegli «Crea politica».	DBA

Operazione	Description	Competenze richieste
Crea un ruolo IAM per consentire l'operazione UNLOAD per Amazon Redshift.	Apri la console IAM e scegli «Ruoli». Scegli «Crea ruolo» e scegli «Servizio AWS» in «Seleziona il tipo di entità affidabile». Scegli «Redshift» per il servizio, scegli «Redshift — Personalizzabile», quindi scegli «Avanti». Scegli la policy «Scarica» che hai creato in precedenza e scegli «Avanti». Inserisci un «Nome ruolo» e scegli «Crea ruolo».	DBA
Associa il ruolo IAM al cluster Amazon Redshift.	Apri la console Amazon Redshift e scegli «Gestisci ruoli IAM». Scegli «Ruoli disponibili» dal menu a discesa e scegli il ruolo che hai creato in precedenza. Scegli «Applica modifiche». Quando lo «Stato» del ruolo IAM in «Gestisci i ruoli IAM» è visualizzato come «In-Sync» , puoi eseguire il comando UNLOAD.	DBA
Interrompi le operazioni di scrittura sul cluster Amazon Redshift.	Devi ricordarti di interrompere tutte le operazioni di scrittura sul cluster Amazon Redshift di origine fino al completamento della migrazione.	DBA

Prepararsi alla migrazione nella regione di destinazione

Operazione	Description	Competenze richieste
Avvia e configura un' EC2 istanza nella regione di destinazione.	Accedi alla Console di gestione AWS per una regione in Cina, Pechino o Ningxia. Dalla EC2 console Amazon, scegli «Launch instance» e crea e configura un' EC2 istanza. Importante: assicurat i che i gruppi EC2 di sicurezza Amazon per le regole in entrata consentano l'accesso illimitato alla porta TCP 22 dal tuo computer di origine. Per ulteriori istruzioni su come avviare e configurare un' EC2 istanza, consulta la sezione «Risorse correlate».	DBA
Registra i dettagli del cluster Amazon Redshift.	Apri la console Amazon Redshift e scegli «Clusters » nel pannello di navigazio ne. Quindi scegli il nome del cluster Amazon Redshift dall'elenco. Nella scheda «Proprietà», nella sezione «Configurazioni del database» , registra il «Nome del database» e «Porta». <databasename>Apri la sezione «Dettagli di connessio ne» e registra l' "Endpoint», che è nel formato «endpoint :<port>/». Importante: assicurati che i gruppi di	DBA

Operazione	Description	Competenze richieste
	sicurezza di Amazon Redshift per le regole in entrata consentano l'accesso illimitato alla porta TCP 5439 dalla tua istanza. EC2	
Connect psql al cluster Amazon Redshift.	<userid><databasename><port>Al prompt dei comandi, specifica le informazioni di connessione eseguendo il comando «psql -h -U -d <endpoint>-p». <userid>Alla richiesta della password psql, inserisci la password per l'utente "». Verrai quindi connesso al cluster Amazon Redshift e potrai inserire comandi in modo interattivo.	DBA
Crea un bucket S3.	Apri la console Amazon S3 e crea un bucket S3 per contenere i file esportati da Amazon Redshift. Per assistenza su questa e altre storie, consulta la sezione «Risorse correlate».	DBA

Operazione	Description	Competenze richieste
Crea una policy IAM che supporti la copia dei dati.	Apri la console IAM e scegli «Policies». Scegli «Crea policy» e scegli la scheda «JSON». Copia e incolla la politica IAM per la copia dei dati dalla sezione «Informaz ioni aggiuntive». Importante: sostituisci «s3_bucket_name» con il nome del tuo bucket S3. Scegli «Rivedi la politica» , inserisci un nome e una descrizione per la politica. Scegli «Crea politica».	DBA
Crea un ruolo IAM per consentire l'operazione COPY per Amazon Redshift.	Apri la console IAM e scegli «Ruoli». Scegli «Crea ruolo» e scegli «Servizio AWS» in «Seleziona il tipo di entità affidabile». Scegli «Redshift» per il servizio, scegli «Redshift — Personalizzabile», quindi scegli «Avanti». Scegli la policy «Copia» che hai creato in precedenza e scegli «Avanti». Inserisci un «Nome ruolo» e scegli «Crea ruolo».	DBA

Operazione	Description	Competenze richieste
Associa il ruolo IAM al cluster Amazon Redshift.	Apri la console Amazon Redshift e scegli «Gestisci ruoli IAM». Scegli «Ruoli disponibili» dal menu a discesa e scegli il ruolo che hai creato in precedenza. Scegli «Applica modifiche». Quando lo «Stato» del ruolo IAM in «Gestisci i ruoli IAM» è visualizzato come «In-Sync» , puoi eseguire il comando «COPY».	DBA

Verifica i dati di origine e le informazioni sugli oggetti prima di iniziare la migrazione

Operazione	Description	Competenze richieste
Verifica le righe nelle tabelle Amazon Redshift di origine.	Utilizza gli script nella sezione «Informazioni aggiuntive» per verificare e registrar e il numero di righe nelle tabelle Amazon Redshift di origine. Ricorda di dividere i dati in modo uniforme per gli script UNLOAD e COPY. Ciò migliorerà l'efficienza di scaricamento e caricamento dei dati, poiché la quantità di dati coperta da ogni script sarà bilanciata.	DBA
Verifica il numero di oggetti di database nel cluster Amazon Redshift di origine.	Utilizza gli script nella sezione «Informazioni aggiuntive» per verificare e registrar	DBA

Operazione	Description	Competenze richieste
	e il numero di database, utenti, schemi, tabelle, viste e funzioni definite dall'utente (UDFs) nel cluster Amazon Redshift di origine.	
Verifica i risultati delle istruzioni SQL prima della migrazione.	Alcune istruzioni SQL per la convalida dei dati devono essere ordinate in base alle situazioni aziendali e relative ai dati effettive. Questo serve a verificare i dati importati per garantire che siano coerenti e visualizzati correttamente.	DBA

Migra dati e oggetti nella regione di destinazione

Operazione	Description	Competenze richieste
Genera script DDL di Amazon Redshift.	Genera script DDL (Data Definition Language) utilizzando i collegamenti dalla sezione «Istruzioni SQL per interrogare Amazon Redshift» nella sezione «Informazioni aggiuntive». Questi script DDL devono includere le query «create user», «create schema», «privilegi sullo schema to user», «create table/view», «privilegi sugli oggetti da usare» e «create function».	DBA

Operazione	Description	Competenze richieste
Crea oggetti nel cluster Amazon Redshift per la regione di destinazione.	Esegui gli script DDL utilizzando l'AWS Command Line Interface (AWS CLI) nella regione AWS in Cina. Questi script creeranno oggetti nel cluster Amazon Redshift per la regione di destinazione.	DBA
Scarica i dati di origine del cluster Amazon Redshift nel bucket S3.	Esegui il comando UNLOAD per scaricare i dati dal cluster Amazon Redshift nella regione di origine nel bucket S3.	DBA, Sviluppatore
Trasferisci i dati del bucket Region S3 di origine nel bucket Region S3 di destinazione.	Trasferisci i dati dal bucket Region S3 di origine al bucket S3 di destinazione. Poiché il comando «$ aws s3 sync» non può essere utilizzato, assicurati di utilizzare la procedura descritta nell'articolo «Trasferimento di dati Amazon S3 dalle regioni AWS alle regioni AWS in Cina» nella sezione «Risorse correlate».	Developer
Carica i dati nel cluster Amazon Redshift di destinazione.	Nello strumento psql per la tua regione di destinazione, esegui il comando COPY per caricare i dati dal bucket S3 al cluster Amazon Redshift di destinazione.	DBA

Verifica i dati nelle regioni di origine e di destinazione dopo la migrazione

Operazione	Description	Competenze richieste
Verifica e confronta il numero di righe nelle tabelle di origine e di destinazione.	Verifica e confronta il numero di righe della tabella nelle regioni di origine e di destinazione per assicurarti che tutte vengano migrate.	DBA
Verifica e confronta il numero di oggetti del database di origine e di destinazione.	Verifica e confronta tutti gli oggetti del database nelle regioni di origine e di destinazione per assicurarti che tutti vengano migrati.	DBA
Verifica e confronta i risultati degli script SQL nelle regioni di origine e di destinazione.	Esegui gli script SQL preparati prima della migrazione. Verifica e confronta i dati per assicurarti che i risultati SQL siano corretti.	DBA
Reimposta le password di tutti gli utenti nel cluster Amazon Redshift di destinazione.	Una volta completata la migrazione e verificato tutti i dati, devi reimpostare tutte le password utente per il cluster Amazon Redshift nella regione AWS in Cina.	DBA

Risorse correlate

- Trasferimento di dati Amazon S3 dalle regioni AWS alle regioni AWS in Cina
- Creazione di un bucket S3
- Reimpostazione di una password utente Amazon Redshift
- documentazione psql

Informazioni aggiuntive

Politica IAM per lo scarico dei dati

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": ["s3:ListBucket"],
      "Resource": ["arn:aws:s3:::s3_bucket_name"]
    },
    {
      "Effect": "Allow",
      "Action": ["s3:GetObject", "s3:DeleteObject"],
      "Resource": ["arn:aws:s3:::s3_bucket_name/*"]
    }
  ]
}
```

Politica IAM per la copia dei dati

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": ["s3:ListBucket"],
      "Resource": ["arn:aws:s3:::s3_bucket_name"]
    },
    {
      "Effect": "Allow",
      "Action": ["s3:GetObject"],
      "Resource": ["arn:aws:s3:::s3_bucket_name/*"]
    }
  ]
}
```

Istruzioni SQL per interrogare Amazon Redshift

```
##Database

select * from pg_database where datdba>1;
```

```
##User

select * from pg_user where usesysid>1;

##Schema

SELECT n.nspname AS "Name",

  pg_catalog.pg_get_userbyid(n.nspowner) AS "Owner"

FROM pg_catalog.pg_namespace n

WHERE n.nspname !~ '^pg_' AND n.nspname <> 'information_schema'

ORDER BY 1;

##Table

select count(*) from pg_tables where schemaname not in
 ('pg_catalog','information_schema');

select schemaname,count(*) from pg_tables where schemaname not in
 ('pg_catalog','information_schema') group by schemaname order by 1;

##View

SELECT

    n.nspname AS schemaname,c.relname AS
 viewname,pg_catalog.pg_get_userbyid(c.relowner) as "Owner"

FROM

    pg_catalog.pg_class AS c

INNER JOIN

    pg_catalog.pg_namespace AS n

    ON c.relnamespace = n.oid

WHERE relkind = 'v' and n.nspname not in ('information_schema','pg_catalog');
```

```
##UDF

SELECT

   n.nspname AS schemaname,

   p.proname AS proname,

   pg_catalog.pg_get_userbyid(p.proowner) as "Owner"

FROM pg_proc p

LEFT JOIN pg_namespace n on n.oid = p.pronamespace

WHERE p.proowner != 1;
```

Script SQL per generare istruzioni DDL

- Script get_schema_priv_by_user
- Genera script TBL_DDL
- Generate_view_ddl
- Genera user_grant_revoke_ddl
- Genera udf_ddl

Trasporta i database PostgreSQL tra due istanze DB Amazon RDS utilizzando pg_transport

Raunak Rishabh e Jitender Kumar, Amazon Web Services

Riepilogo

Questo modello descrive i passaggi per la migrazione di database estremamente grandi tra due istanze DB di Amazon Relational Database Service (Amazon RDS) per PostgreSQL utilizzando l'estensione pg_transport. Questa estensione fornisce un meccanismo di trasporto fisico per spostare i singoli database. Trasmettendo i file di database con un'elaborazione minima, fornisce un metodo estremamente veloce per migrare database di grandi dimensioni tra istanze DB con tempi di inattività minimi. Questa estensione utilizza un modello pull in cui l'istanza DB di destinazione importa il database dall'istanza DB di origine.

Prerequisiti e limitazioni

Prerequisiti

- Entrambe le istanze DB devono eseguire la stessa versione principale di PostgreSQL.
- Il database non deve esistere sulla destinazione. In caso contrario, il trasporto non riesce.
- Nessuna estensione diversa da pg_transport deve essere abilitata nel database di origine.
- Tutti gli oggetti del database di origine devono trovarsi nel tablespace pg_default predefinito.
- Il gruppo di sicurezza dell'istanza DB di origine dovrebbe consentire il traffico proveniente dall'istanza DB di destinazione.
- Installa un client PostgreSQL come psql o utilizza l'istanza database Amazon PgAdminRDS PostgreSQL. Puoi installare il client nel tuo sistema locale o utilizzare un'istanza Amazon Elastic Compute Cloud (Amazon EC2). In questo modello, utilizziamo psql su un' EC2 istanza.

Limitazioni

- Non puoi trasportare database tra diverse versioni principali di Amazon RDS for PostgreSQL.
- I privilegi di accesso e la proprietà dal database di origine non vengono trasferiti al database di destinazione.
- Non è possibile trasportare database su repliche di lettura o su istanze principali di repliche di lettura.

- Non è possibile utilizzare i tipi di dati reg in nessuna tabella di database che si prevede di trasportare con questo metodo.
- È possibile eseguire fino a 32 trasporti totali (incluse importazioni ed esportazioni) contemporaneamente su un'istanza DB.
- Non è possibile rinominare le tabelle. include/exclude Tutto viene migrato così com'è.

Attenzione

- Eseguite dei backup prima di rimuovere l'estensione, poiché rimuovendo l'estensione vengono rimossi anche gli oggetti dipendenti e alcuni dati fondamentali per il funzionamento del database.
- Considera la classe di istanza e i processi in esecuzione su altri database sull'istanza di origine quando determini il numero di worker e `work_mem` i valori per pg_transport.
- All'avvio del trasporto, tutte le connessioni sul database di origine vengono interrotte e il database viene messo in modalità di sola lettura.

> **Note**
>
> Quando il trasporto è in esecuzione su un database, non influisce sugli altri database sullo stesso server.

Versioni del prodotto

- Amazon RDS per PostgreSQL 10.10 e versioni successive e Amazon RDS for PostgreSQL 11.5 e versioni successive. Per informazioni sulla versione più recente, consulta Trasporto di database PostgreSQL tra istanze DB nella documentazione di Amazon RDS.

Architecture

Tools (Strumenti)

- pg_transport fornisce un meccanismo di trasporto fisico per spostare ogni database. Tramite lo streaming dei file di database con un'elaborazione minima, il trasporto fisico sposta i dati molto più velocemente rispetto ai tradizionali processi di dump e load e richiede tempi di inattività minimi. Transportable Database di PostgreSQL utilizza un modello pull in cui l'istanza database

di destinazione importa il database dall'istanza database di origine. Questa estensione viene installata sulle istanze DB quando si preparano gli ambienti di origine e di destinazione, come spiegato in questo schema.

- psql ti consente di connetterti e lavorare con le tue istanze DB PostgreSQL. Per installare psql sul tuo sistema, consulta la pagina dei download di PostgreSQL.

Epiche

Crea il gruppo di parametri di destinazione

Operazione	Description	Competenze richieste
Crea un gruppo di parametri per il sistema di destinazione.	Specificate un nome di gruppo che lo identifichi come gruppo di parametri di destinazione; ad esempio,`pgtarget-param-group` . Per istruzioni, consulta la documentazione di Amazon RDS.	DBA
Modificate i parametri per il gruppo di parametri.	Imposta i seguenti parametri: 1. Aggiungi `pg_transport` al `shared_preload_libraries` parametro. `shared_preload_libraries = pg_stat_statements, pg_transport` 2. Impostare il parametro `pg_transport.num_workers` . Scegli il numero di lavoratori con cui desideri eseguire il trasporto. Il valore impostato determina	DBA

Operazione	Description	Competenze richieste
	il numero di `transport.send_file` lavoratori che verranno creati nell'origine. 3. Aumentate il valore di `max_worker_processes` a più di tre volte il valore di`pg_transport.num_workers` . Ad esempio, se si imposta il valore `pg_transport.num_workers` su 4, il `max_worker_processes` valore deve essere almeno 13. Se ciò fallisce, pg_transport consiglia un valore minimo. 4. Impostato su 1`pg_transport.timing` . Questa impostazione consente la segnalazione delle informazioni sulla tempistica durante il trasporto. 5. Impostare il parametro `pg_transport.work_mem` . Questo parametro specifica la memoria massima da allocare a ciascun lavoratore. Il valore predefinito è 128 MB. Per ulteriori informazioni su questi parametri, consulta la	

Operazione	Description	Competenze richieste
	documentazione di Amazon RDS.	

Crea il gruppo di parametri di origine

Operazione	Description	Competenze richieste
Crea un gruppo di parametri per il sistema di origine.	Specificate un nome di gruppo che lo identifichi come gruppo di parametri di origine; ad esempio,`pgsource-param-group` . Per istruzioni, consulta la documentazione di Amazon RDS.	DBA
Modificate i parametri per il gruppo di parametri.	Imposta i seguenti parametri: 1. Aggiungi `pg_transport` al `shared_preload_libraries` parametro. `shared_preload_libraries = pg_stat_statements, pg_transport` 2. Impostare il parametro `pg_transport.num_workers` . Il valore di questo parametro definito nell'obiettivo determina il numero di `transport.send_file` lavoratori da utilizzare. Se hai un'importazione in esecuzione su questa	DBA

Operazione	Description	Competenze richieste
	istanza, aumenta questo valore, ma considera il numero di lavoratori già in esecuzione. 3. Aumentate il valore `max_worker_processes` di oltre tre volte il valore dell'`pg_transport.num_workers` obiettivo. Ad esempio, se imposti il valore su 4 sulla destinazione, il `max_worker_processes` valore sull'origine deve essere almeno 13. `pg_transport.num_workers` Se ciò fallisce, pg_transport consiglia un valore minimo. 4. Impostare il parametro `pg_transport.work_mem` . Questo parametro specifica la memoria massima da allocare a ciascun lavoratore. Il valore predefinito è 128 MB. Per ulteriori informazioni su questi parametri, consulta la documentazione di Amazon RDS.	

Prepara l'ambiente di destinazione

Operazione	Description	Competenze richieste
Crea una nuova istanza DB Amazon RDS for PostgreSQL in cui trasportare il database di origine.	Determina la classe dell'istanza e la versione di PostgreSQL in base ai tuoi requisiti aziendali.	DBA, amministratore di sistema, architetto del database
Modifica il gruppo di sicurezza della destinazione per consentire le connessioni sulla porta dell'istanza DB dall' EC2 istanza.	Per impostazione predefini ta, la porta per l'istanza PostgreSQL è 5432. Se stai usando un'altra porta, le connessioni a quella porta devono essere aperte per l'istanza. EC2	DBA, amministratore di sistema
Modifica l'istanza e assegna il nuovo gruppo di parametri di destinazione.	Ad esempio, `pgtarget-param-group` .	DBA
Riavvia l'istanza database Amazon RDS di destinazione.	I parametri `shared_preload_libraries` e `max_worker_processes` sono parametri statici e richiedono il riavvio dell'istanza.	DBA, amministratore di sistema
Connect al database dall' EC2 istanza usando psql.	Utilizza il comando : `psql -h <rds_end_point> -p PORT -U username -d database -W`	DBA
Crea l'estensione pg_transport.	Esegui la seguente query come utente con il ruolo: `rds_superuser`	DBA

Operazione	Description	Competenze richieste
	`create extension pg_transport;`	

Preparare l'ambiente di origine

Operazione	Description	Competenze richieste
Modifica il gruppo di sicurezza dell'origine per consentire le connessioni sulla porta dell'istanza DB dall'istanza Amazon e dall' EC2 istanza DB di destinazione	Per impostazione predefinita, la porta per l'istanza PostgreSQL è 5432. Se stai usando un'altra porta, le connessioni a quella porta devono essere aperte per l'istanza. EC2	DBA, amministratore di sistema
Modifica l'istanza e assegna il nuovo gruppo di parametri di origine.	Ad esempio, `pgsource-param-group` .	DBA
Riavvia l'istanza database Amazon RDS di origine.	I parametri `shared_preload_libraries` e `max_worker_processes` sono parametri statici e richiedono il riavvio dell'istanza.	DBA
Connect al database dall' EC2 istanza usando psql.	Utilizza il comando : `psql -h <rds_end_point> -p PORT -U username -d database -W`	DBA
Crea l'estensione pg_transport e rimuovi tutte le altre	Il trasporto avrà esito negativo se nel database di origine sono installate estension	DBA

Operazione	Description	Competenze richieste
estensioni dai database da trasportare.	i diverse da pg_transport. Questo comando deve essere eseguito da un utente con il ruolo. `rds_superuser`	

Esegui il trasporto

Operazione	Description	Competenze richieste
Esegui una corsa a secco.	Usa la `transport.import_from_server` funzione per eseguire prima una corsa a secco: `SELECT transport.import_from_server( 'source-db-instance-endpoint', source-db-instance-port, 'source-db-instance-user', 'source-user-password', 'source-database-name', 'destination-user-password', 'true');` L'ultimo parametro di questa funzione (impostato su`true`) definisce il funzionamento a secco. Questa funzione visualizza tutti gli errori che si possono verificare durante l'esecuzione del trasporto principale.	DBA

Operazione	Description	Competenze richieste
	Risolvete gli errori prima di eseguire il trasporto principale.	
Se l'esecuzione a secco ha esito positivo, avvia il trasporto del database.	Esegui la `transport.import_from_server` funzione per eseguire il trasporto. Si collega alla fonte e importa i dati. `SELECT transport.import_from_server( 'source-db-instance-endpoint', source-db-instance-port, 'source-db-instance-user', 'source-user-password', 'source-database-name', 'destination-user-password', false);` L'ultimo parametro di questa funzione (impostato su`false`) indica che non si tratta di un funzionamento a secco.	DBA

Operazione	Description	Competenze richieste
Eseguire le fasi successive al trasporto.	Una volta completato il trasporto del database: • Convalida i dati nell'ambiente di destinazione. • Aggiungi tutti i ruoli e le autorizzazioni alla destinazione. • Abilita tutte le estensioni richieste nella destinazione e nell'origine, se necessario. • Ripristina il valore del `max_worker_processes` parametro.	DBA

Risorse correlate

- Documentazione Amazon RDS
- documentazione pg_transport
- Migrazione di database utilizzando database trasportabili RDS PostgreSQL (post sul blog)
- Download PostgreSQL
- utilità psql
- Creazione di un gruppo di parametri database
- Modifica i parametri in un gruppo di parametri DB
- Download PostgreSQL

Conversione piattaforma

Argomenti

- Esportazione di un database Microsoft SQL Server in Amazon S3 utilizzando AWS DMS
- Esegui la migrazione dei valori Oracle CLOB su singole righe in PostgreSQL su AWS
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for Oracle utilizzando l'importazione diretta di Oracle Data Pump tramite un collegamento al database
- Esegui la migrazione PeopleSoft da Oracle ad Amazon RDS Custom
- Esegui la migrazione della funzionalità Oracle ROWID a PostgreSQL su AWS
- Esegui la migrazione dei codici di errore del database Oracle a un database compatibile con Amazon Aurora PostgreSQL
- Esegui la migrazione di SAP ASE su Amazon Aurora, compatibile EC2 con PostgreSQL utilizzando AWS SCT e AWS DMS
- Migrazione dei certificati SSL di Windows su un Application Load Balancer utilizzando ACM
- Esegui la migrazione di una coda di messaggistica da Microsoft Azure Service Bus ad Amazon SQS
- Esegui la migrazione di un database Oracle JD Edwards EnterpriseOne su AWS utilizzando Oracle Data Pump e AWS DMS
- Esegui la migrazione di un PeopleSoft database Oracle su AWS utilizzando AWS DMS
- Esegui la migrazione di un database MySQL locale su Amazon RDS for MySQL
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon RDS for SQL Server
- Esegui la migrazione dei dati da Microsoft Azure Blob ad Amazon S3 utilizzando Rclone
- Migrazione da Couchbase Server a Couchbase Capella su AWS
- Esegui la migrazione da IBM WebSphere Application Server ad Apache Tomcat su Amazon EC2
- Esegui la migrazione da IBM WebSphere Application Server ad Apache Tomcat su Amazon EC2 con Auto Scaling
- Esegui la migrazione di un'applicazione.NET da Microsoft Azure App Service ad AWS Elastic Beanstalk
- Esegui la migrazione da Oracle WebLogic ad Apache Tomcat (ToMee) su Amazon ECS
- Esegui la migrazione di un database Oracle da Amazon EC2 ad Amazon RDS for Oracle utilizzando AWS DMS

- Esegui la migrazione di un database Oracle locale ad Amazon OpenSearch Service utilizzando Logstash
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for Oracle
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for Oracle utilizzando Oracle Data Pump
- Esegui la migrazione da PostgreSQL su Amazon RDS per PostgreSQL utilizzando EC2 pglogical
- Esegui la migrazione di un database PostgreSQL locale su Aurora PostgreSQL
- Esegui la migrazione di un database Microsoft SQL Server locale a Microsoft SQL Server su Amazon EC2 con Linux
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon RDS for SQL Server utilizzando server collegati
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon RDS for SQL Server utilizzando metodi di backup e ripristino nativi
- Esegui la migrazione di un database Microsoft SQL Server su Aurora MySQL utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database MariaDB locale su Amazon RDS for MariaDB utilizzando strumenti nativi
- Esegui la migrazione di un database MySQL locale su Aurora MySQL
- Esegui la migrazione dei database MySQL locali su Aurora MySQL utilizzando Percona, Amazon EFS e Amazon S3 XtraBackup
- Esegui la migrazione di applicazioni Java locali su AWS utilizzando AWS App2Container
- Migrazione di file system condivisi in una migrazione di AWS grandi dimensioni
- Esegui la migrazione di un database Oracle ad Amazon RDS for Oracle utilizzando gli adattatori flat file GoldenGate Oracle
- Modifica le applicazioni Python e Perl per supportare la migrazione dei database da Microsoft SQL Server a Amazon Aurora PostgreSQL Compatible Edition

Esportazione di un database Microsoft SQL Server in Amazon S3 utilizzando AWS DMS

Sweta Krishna, Amazon Web Services

Riepilogo

Le organizzazioni spesso devono copiare i database su Amazon Simple Storage Service (Amazon S3) per la migrazione dei database, il backup e il ripristino, l'archiviazione dei dati e l'analisi dei dati. Questo modello descrive come esportare un database Microsoft SQL Server in Amazon S3. Il database di origine può essere ospitato in locale o su Amazon Elastic Compute Cloud (Amazon EC2) o Amazon Relational Database Service (Amazon RDS) per Microsoft SQL Server sul cloud Amazon Web Services (AWS).

I dati vengono esportati utilizzando AWS Database Migration Service (AWS DMS). Per impostazione predefinita, AWS DMS scrive dati CDC (full load and change data capture) in formato con valori separati da virgole (.csv). Per uno storage più compatto e opzioni di interrogazione più veloci, questo modello utilizza l'opzione di formato Apache Parquet (.parquet).

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un ruolo AWS Identity and Access Management (IAM) per l'account con accesso in scrittura, eliminazione e tag al bucket S3 di destinazione e AWS DMS (`dms.amazonaws.com`) aggiunto come entità affidabile a questo ruolo IAM
- Un database Microsoft SQL Server locale (o Microsoft SQL Server su un' EC2 istanza o un database Amazon RDS for SQL Server)
- Connettività di rete tra il cloud privato virtuale (VPC) su AWS e la rete locale fornita da AWS Direct Connect o una rete privata virtuale (VPN)

Limitazioni

- Un bucket S3 abilitato per VPC (gateway VPC) non è attualmente supportato nelle versioni di AWS DMS precedenti alla 3.4.7.
- Le modifiche apportate alla struttura della tabella di origine durante il caricamento completo non sono supportate.

- La modalità LOB (full large binary object) di AWS DMS non è supportata.

Versioni del prodotto

- Microsoft SQL Server versioni 2005 o successive per le edizioni Enterprise, Standard, Workgroup e Developer.
- Il supporto per Microsoft SQL Server versione 2019 come sorgente è disponibile nelle versioni 3.3.2 e successive di AWS DMS.

Architecture

Stack tecnologico di origine

- Un database Microsoft SQL Server locale (o Microsoft SQL Server su un' EC2 istanza o un database Amazon RDS for SQL Server)

Stack tecnologico Target

- AWS Direct Connect
- AWS DMS
- Simple Storage Service (Amazon S3)

Architettura di destinazione

Tools (Strumenti)

- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali.
- AWS Direct Connect collega la rete interna a una posizione Direct Connect tramite un cavo Ethernet standard in fibra ottica. Con questa connessione, puoi creare interfacce virtuali direttamente ai servizi AWS pubblici bypassando i provider di servizi Internet nel tuo percorso di rete.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Epiche

Preparati per la migrazione

Operazione	Description	Competenze richieste
Convalida la versione del database.	Convalida la versione del database di origine e assicurati che sia supportata da AWS DMS. Per informazioni sulle versioni supportate del database SQL Server, consulta Using a Microsoft SQL Server database as a source for AWS DMS.	DBA
Crea un VPC e un gruppo di sicurezza.	Nel tuo account AWS, crea un VPC e un gruppo di sicurezza. Per ulteriori informazioni, consulta la documentazione di Amazon VPC.	Amministratore di sistema
Crea un utente per il task AWS DMS.	Crea un utente AWS DMS nel database di origine e concedigli le autorizzazioni READ. Questo utente verrà utilizzato da AWS DMS.	DBA
Verifica la connettività DB.	Verifica la connettività all'istanza DB di SQL Server dall'utente AWS DMS.	DBA
Crea un bucket S3.	Crea il bucket S3 di destinazione. Questo bucket conterrà i dati della tabella migrati.	Amministratore di sistema
Crea una politica e un ruolo IAM.	1. Per creare una policy IAM con autorizzazioni bucket,	Amministratore di sistema

Operazione	Description	Competenze richieste
	usa il codice nella sezione Informazioni aggiuntive. 2. Crea il ruolo per AWS DMS e associa la policy al ruolo.	

Migra i dati utilizzando AWS DMS

Operazione	Description	Competenze richieste
Creare un'istanza di replica di AWS DMS.	Accedi alla Console di gestione AWS e apri la console AWS DMS. Nel pannello di navigazione, scegli Istanze di replica, Crea istanza di replica. Per istruzioni, consulta il passaggio 1 nella documentazione di AWS DMS.	DBA
Crea endpoint di origine e destinazione.	Crea endpoint di origine e destinazione. Verifica la connessione dall'istanza di replica agli endpoint di origine e di destinazione. Per istruzioni, consulta il passaggio 2 nella documentazione di AWS DMS.	DBA
Creare un'attività di replica.	Crea un'attività di replica e seleziona Full load o full load with change data capture (CDC) per migrare i dati da SQL Server al bucket S3. Per istruzioni, consulta il	DBA

Operazione	Description	Competenze richieste
	passaggio 3 nella documenta zione di AWS DMS.	
Avvia la replica dei dati.	Avvia l'attività di replica e monitora i log per eventuali errori.	DBA

Convalida i dati

Operazione	Description	Competenze richieste
Convalida i dati migrati.	Sulla console, accedi al bucket S3 di destinazione. Apri la sottocartella con lo stesso nome del database di origine. Verifica che la cartella contenga tutte le tabelle che sono state migrate dal database di origine.	DBA

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Chiudi ed elimina le risorse AWS temporanee.	Chiudi le risorse AWS temporanee che hai creato per la migrazione dei dati, come l'istanza di replica AWS DMS, ed eliminale dopo aver convalidato l'esportazione.	DBA

Risorse correlate

- Guida per l'utente di AWS Database Migration Service

- Utilizzo di un database Microsoft SQL Server come origine per AWS DMS
- Utilizzo di Amazon S3 come destinazione per AWS Database Migration Service
- Utilizzo di un bucket S3 come destinazione AWS DMS (AWS re:Post)

Informazioni aggiuntive

Utilizza il codice seguente per aggiungere una policy IAM con autorizzazioni per i bucket S3 per il ruolo AWS DMS. Sostituisci `bucketname` con il nome del tuo bucket.

```
{
     "Version": "2012-10-17",
     "Statement": [
         {
             "Effect": "Allow",
             "Action": [
                 "s3:PutObject",
                 "s3:DeleteObject"
             ],
             "Resource": [
                 "arn:aws:s3:::bucketname*"
             ]
         },
         {
             "Effect": "Allow",
             "Action": [
                 "s3:ListBucket"
             ],
             "Resource": [
                 "arn:aws:s3:::bucketname*"
             ]
         }
     ]
 }
```

Esegui la migrazione dei valori Oracle CLOB su singole righe in PostgreSQL su AWS

Sai Krishna Namburu e Sindhusha Paturu, Amazon Web Services

Riepilogo

Questo modello descrive come suddividere i valori di Oracle Character Large Object (CLOB) in singole righe in Amazon Aurora PostgreSQL Compatible Edition e Amazon Relational Database Service (Amazon RDS) per PostgreSQL. PostgreSQL non supporta il tipo di dati CLOB.

Le tabelle con partizioni a intervalli vengono identificate nel database Oracle di origine e il nome della tabella, il tipo di partizione, l'intervallo della partizione e altri metadati vengono acquisiti e caricati nel database di destinazione. Puoi caricare dati CLOB di dimensioni inferiori a 1 GB nelle tabelle di destinazione come testo utilizzando AWS Database Migration Service (AWS DMS) oppure puoi esportare i dati in formato CSV, caricarli in un bucket Amazon Simple Storage Service (Amazon S3) e migrarli nel database PostgreSQL di destinazione.

Dopo la migrazione, puoi utilizzare il codice PostgreSQL personalizzato fornito con questo modello per dividere i dati CLOB in singole righe in base al nuovo identificatore `CHR(10)` di caratteri di riga () e compilare la tabella di destinazione.

Prerequisiti e limitazioni

Prerequisiti

- Una tabella di database Oracle con partizioni e record a intervalli con un tipo di dati CLOB.
- Un database compatibile con Aurora PostgreSQL o Amazon RDS for PostgreSQL con una struttura di tabella simile alla tabella di origine (stesse colonne e tipi di dati).

Limitazioni

- Il valore CLOB non può superare 1 GB.
- Ogni riga della tabella di destinazione deve avere un nuovo identificatore di carattere di riga.

Versioni del prodotto

- Oracle 12c
- Aurora Postgres 11.6

Architecture

Il diagramma seguente mostra una tabella Oracle di origine con dati CLOB e la tabella PostgreSQL equivalente nella versione 11.6 compatibile con Aurora PostgreSQL.

Tools (Strumenti)

Servizi AWS

- Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL.
- Amazon Relational Database Service (Amazon RDS) per PostgreSQL ti aiuta a configurare, gestire e scalare un database relazionale PostgreSQL nel cloud AWS.
- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri strumenti

Puoi utilizzare i seguenti strumenti client per connetterti, accedere e gestire i tuoi database Aurora compatibili con PostgreSQL e Amazon RDS for PostgreSQL. (Questi strumenti non vengono utilizzati all'interno di questo schema).

- pgAdmin è uno strumento di gestione open source per PostgreSQL. Fornisce un'interfaccia grafica che consente di creare, gestire e utilizzare oggetti di database.
- DBeaverè uno strumento di database open source per sviluppatori e amministratori di database. È possibile utilizzare lo strumento per manipolare, monitorare, analizzare, amministrare e migrare i dati.

Best practice

Per le best practice per la migrazione del database da Oracle a PostgreSQL, consulta il post sul blog AWS Best practice for migrating an Oracle database to Amazon RDS PostgreSQL o Amazon Aurora PostgreSQL: considerazioni sul processo di migrazione e sull'infrastruttura.

Per le best practice per la configurazione del task AWS DMS per la migrazione di oggetti binari di grandi dimensioni, consulta Migrating large binary objects () LOBs nella documentazione di AWS DMS.

Epiche

Identifica i dati CLOB

Operazione	Description	Competenze richieste
Analizza i dati CLOB.	Nel database Oracle di origine, analizza i dati CLOB per vedere se contengono intestazioni di colonna, in modo da poter determinare il metodo di caricamento dei dati nella tabella di destinazione. Per analizzare i dati di input, utilizzare la seguente query. `SELECT * FROM clobdata_or;`	Developer
Caricate i dati CLOB nel database di destinazione.	Esegui la migrazione della tabella con dati CLOB a una tabella provvisoria (staging) nel database di destinazione Aurora o Amazon RDS. Puoi utilizzare AWS DMS o caricare i dati come file CSV in un bucket Amazon S3. Per informazioni sull'utilizzo di AWS DMS per questa attività, consulta Utilizzo di un database Oracle come origine e Utilizzo di un database PostgreSQL come destinazi	Ingegnere addetto alla migrazione, DBA

Operazione	Description	Competenze richieste
	one nella documentazione di AWS DMS. Per informazioni sull'utilizzo di Amazon S3 per questa attività, consulta Using Amazon S3 come target nella documentazione di AWS DMS.	

Convalida la tabella PostgreSQL di destinazione.	Convalida i dati di destinazione, incluse le intestazioni, rispetto ai dati di origine utilizzando le seguenti query nel database di destinazione.
	
	`SELECT * FROM clobdata_pg;`
	`SELECT * FROM clobdatatarget;`
	
	Confronta i risultati con i risultati delle query del database di origine (dal primo passaggio).
		Developer

Dividi i dati CLOB in righe separate.	Esegui il codice PostgreSQL personalizzato fornito nella sezione Informazioni aggiuntive per dividere i dati CLOB e inserirli in righe separate nella tabella PostgreSQL di destinazione.	Developer

Convalida i dati.

Operazione	Description	Competenze richieste

Convalida i dati nella tabella di destinazione.	Convalida i dati inseriti nella tabella di destinazione utilizzando le seguenti query.
	`SELECT * FROM clobdata_pg;`
	`SELECT * FROM clobdatatarget;`
		Developer

Risorse correlate

- Tipo di dati CLOB (documentazione Oracle)
- Tipi di dati (documentazione PostgreSQL)

Informazioni aggiuntive

Funzione PostgreSQL per la suddivisione dei dati CLOB

```
do
$$
declare
totalstr varchar;
str1 varchar;
str2 varchar;
pos1 integer := 1;
pos2 integer ;
len integer;


begin
        select rawdata||chr(10) into totalstr from clobdata_pg;
        len :=  length(totalstr) ;
        raise notice 'Total length : %',len;
        raise notice 'totalstr : %',totalstr;
        raise notice 'Before while loop';
```

```
        while pos1 < len  loop


                  select position (chr(10) in totalstr) into pos2;
                 raise notice '1st position of new line : %',pos2;


                str1 := substring (totalstr,pos1,pos2-1);
                raise notice 'str1 : %',str1;


                   insert into clobdatatarget(data) values (str1);
                   totalstr := substring(totalstr,pos2+1,len);
                   raise notice 'new totalstr :%',totalstr;
                len :=  length(totalstr) ;


        end loop;
end
$$
LANGUAGE 'plpgsql' ;
```

Esempi di input e output

Puoi usare i seguenti esempi per provare il codice PostgreSQL prima di migrare i tuoi dati.

Crea un database Oracle con tre righe di input.

```
CREATE TABLE clobdata_or (
id INTEGER GENERATED ALWAYS AS IDENTITY,
rawdata clob  );


insert into clobdata_or(rawdata) values (to_clob('test line 1') || chr(10) ||
 to_clob('test line 2') || chr(10) || to_clob('test line 3') || chr(10));
COMMIT;


SELECT * FROM clobdata_or;
```

Viene visualizzato il seguente output.

id	dati grezzi
1	linea di test 1 linea di test 2 linea di test 3

Carica i dati di origine in una tabella intermedia PostgreSQL () per l'elaborazione. `clobdata_pg`

```
SELECT * FROM clobdata_pg;

CREATE TEMP TABLE clobdatatarget (id1 SERIAL,data VARCHAR );

<Run the code in the additional information section.>

SELECT * FROM clobdatatarget;
```

Questo mostra il seguente output.

id1	data
1	linea di test 1
2	linea di test 2
3	linea di test 3

Esegui la migrazione di un database Oracle locale ad Amazon RDS for Oracle utilizzando l'importazione diretta di Oracle Data Pump tramite un collegamento al database

Rizwan Wangde, Amazon Web Services

Riepilogo

Numerosi modelli riguardano la migrazione dei database Oracle locali ad Amazon Relational Database Service (Amazon RDS) per Oracle utilizzando Oracle Data Pump, un'utilità Oracle nativa che rappresenta il modo preferito per migrare carichi di lavoro Oracle di grandi dimensioni. Questi modelli prevedono in genere l'esportazione di schemi o tabelle di applicazioni in file di dump, il trasferimento dei file di dump in una directory di database su Amazon RDS for Oracle e quindi l'importazione degli schemi applicativi e dei dati dai file di dump.

Utilizzando questo approccio, una migrazione può richiedere più tempo a seconda della dimensione dei dati e del tempo necessario per trasferire i file di dump sull'istanza Amazon RDS. Inoltre, i file di dump risiedono nel volume Amazon Elastic Block Store (Amazon EBS) dell'istanza Amazon RDS, che deve essere sufficientemente grande per il database e i file di dump. Quando i file di dump vengono eliminati dopo l'importazione, lo spazio vuoto non può essere recuperato, quindi continui a pagare per lo spazio inutilizzato.

Questo modello mitiga questi problemi eseguendo un'importazione diretta sull'istanza Amazon RDS utilizzando l'API Oracle Data Pump (`DBMS_DATAPUMP`) su un collegamento al database. Il pattern avvia una pipeline di esportazione e importazione simultanea tra i database di origine e di destinazione. Questo modello non richiede il dimensionamento di un volume EBS per i file di dump perché nessun file di dump viene creato o memorizzato nel volume. Questo approccio consente di risparmiare il costo mensile dello spazio su disco inutilizzato.

Prerequisiti e limitazioni

Prerequisiti

- Un account Amazon Web Services (AWS) attivo.
- Un cloud privato virtuale (VPC) configurato con sottoreti private su almeno due zone di disponibilità, per fornire l'infrastruttura di rete per l'istanza Amazon RDS.
- Un database Oracle in un data center locale o gestito autonomamente su Amazon Elastic Compute Cloud (Amazon). EC2

- Un'istanza Amazon RDS for Oracle esistente in un'unica zona di disponibilità. L'utilizzo di una singola zona di disponibilità migliora le prestazioni di scrittura durante la migrazione. Un'implementazione Multi-AZ può essere abilitata 24-48 ore prima del cutover.

  Questa soluzione può anche utilizzare Amazon RDS Custom for Oracle come destinazione.
- AWS Direct Connect (consigliato per database di grandi dimensioni).
- Connettività di rete e regole firewall in locale configurate per consentire una connessione in entrata dall'istanza Amazon RDS al database Oracle locale.

Limitazioni

- Il limite di dimensione del database su Amazon RDS for Oracle è di 64 tebibyte (TiB) a dicembre 2022.
- La dimensione massima di un singolo file su un'istanza database di Amazon RDS for Oracle è di 16 TiB. Questo è importante da sapere perché potrebbe essere necessario distribuire le tabelle su più tablespace.

Versioni del prodotto

- Database di origine: Oracle Database versione 10g Release 1 e successive.
- Database di destinazione: per l'elenco più recente delle versioni ed edizioni supportate su Amazon RDS, consulta Amazon RDS for Oracle nella documentazione AWS.

Architecture

Stack tecnologico di origine

- Database Oracle autogestito in locale o nel cloud

Stack tecnologico Target

- Amazon RDS per Oracle o Amazon RDS Custom per Oracle

Architettura Target

Il diagramma seguente mostra l'architettura per la migrazione da un database Oracle locale ad Amazon RDS for Oracle in un ambiente Single-AZ. Le direzioni delle frecce illustrano il flusso di dati nell'architettura. Il diagramma non mostra quale componente sta avviando la connessione.

1. L'istanza Amazon RDS for Oracle si connette al database Oracle di origine locale per eseguire una migrazione a pieno carico tramite il collegamento al database.
2. AWS Database Migration Service (AWS DMS) si connette al database Oracle di origine locale per eseguire la replica continua utilizzando Change Data Capture (CDC).
3. Le modifiche CDC vengono applicate al database Amazon RDS for Oracle.

Tools (Strumenti)

Servizi AWS

- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati all'interno Cloud AWS o tra combinazioni di configurazioni cloud e locali. Questo modello utilizza CDC e l'impostazione Replica solo le modifiche ai dati.
- AWS Direct Connectcollega la rete interna a una Direct Connect posizione tramite un cavo Ethernet standard in fibra ottica. Con questa connessione, è possibile creare interfacce virtuali direttamente al pubblico Servizi AWS ignorando i provider di servizi Internet nel percorso di rete.
- Amazon Relational Database Service ti aiuta a configurare, gestire e scalare un database relazionale Oracle nel cloud AWS.

Altri strumenti

- Oracle Data Pump ti aiuta a spostare dati e metadati da un database all'altro a velocità elevate.
- Strumenti client come Oracle Instant Client o SQL Developer vengono utilizzati per connettere ed eseguire query SQL sul database.

Best practice

Sebbene Direct Connect utilizzi connessioni di rete private dedicate tra la rete locale e AWS, considera le seguenti opzioni per una maggiore sicurezza e crittografia dei dati per i dati in transito:

- Una rete privata virtuale (VPN) che utilizza AWS Site-to-Site VPN o una connessione IPsec VPN dalla rete locale alla rete AWS

- Oracle Database Native Network Encryption configurato sul database Oracle locale
- Crittografia tramite TLS

Epiche

Prepara il database Oracle di origine locale

Operazione	Description	Competenze richieste
Configura la connettività di rete dal database di destinazione al database di origine.	Configura la rete e il firewall locali per consentire la connessione in entrata dall'istanza Amazon RDS di destinazione al database Oracle di origine locale.	Amministratore di rete, tecnico della sicurezza

Crea un utente del database con i privilegi appropriati.	Crea un utente del database nel database Oracle di origine locale con i privilegi per migrare i dati tra l'origine e la destinazione utilizzando Oracle Data Pump:
	`GRANT CONNECT to <migration_user>;`
	`GRANT DATAPUMP_EXP_FULL_DATABASE to <migration_user>;`
	`GRANT SELECT ANY TABLE to <migration_user>;`
		DBA

Preparare il database di origine locale per la migrazione al AWS DMS CDC.	(Facoltativo) Preparare il database Oracle di origine locale per la migrazione AWS DMS CDC dopo il completamento di Oracle Data Pump Full Load:	DBA

Operazione	Description	Competenze richieste

	1. Configura i privilegi aggiuntivi necessari per gestire FLASHBACK durante la migrazione di Oracle Data Pump:
	`GRANT FLASHBACK ANY TABLE to <migration_user>;`
	`GRANT FLASHBACK ARCHIVE ADMINISTER to <migration_user>;`
	2. Per configurare i privilegi dell'account utente richiesti su una fonte Oracle autogestita per AWS DMS, consulta la documentazione.AWS DMS
	3. Per preparare un database di origine Oracle autogestito da utilizzare con CDC AWS DMS, consulta la documentazione.AWS DMS	

Installa e configura SQL Developer.	Installa e configura SQL Developer per connettere ed eseguire query SQL sui database di origine e di destinazione.	DBA, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
Genera uno script per creare i tablespace.	Usa la seguente query SQL di esempio per generare lo script nel database di origine: `SELECT 'CREATE TABLESPACE ' tablespace_name ' DATAFILE SIZE 1G AUTOEXTEND ON MAXSIZE UNLIMITED;' from dba_tablespaces where tablespace_name not in ('SYSTEM', 'SYSAUX','TEMP','UNDOTBS1') order by 1;` Lo script verrà applicato al database di destinazione.	DBA
Genera uno script per creare utenti, profili, ruoli e privilegi.	Per generare uno script per creare gli utenti, i profili, i ruoli e i privilegi del database, utilizzare gli script del documento di Oracle Support How to Extract DDL for User, inclusi privilegi e ruoli utilizzando dbms_metadata.get_ddl (Doc ID 2739952.1) (è richiesto un account Oracle). Lo script verrà applicato al database di destinazione.	DBA

Preparare l'istanza Amazon RDS for Oracle di destinazione

Operazione	Description	Competenze richieste

Crea un collegamento al database di origine e verifica la connettività.	Per creare un collegamento al database di origine locale, puoi utilizzare il seguente comando di esempio:
	`CREATE DATABASE LINK link2src CONNECT TO <migration_user_account> IDENTIFIED BY <password> USING '(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=<dns or ip address of remote db>) (PORT=<listener port>))(CONNECT_DATA=(SID=<remote SID>)))';`
	Per verificare la connettività, esegui il seguente comando SQL:
	`select * from dual@link2src;`
	La connettività ha esito positivo se la risposta è X.
		DBA

Esegui gli script per preparare l'istanza di destinazione.	Esegui gli script generati in precedenza per preparare l'istanza Amazon RDS for Oracle di destinazione:	DBA, ingegnere addetto alla migrazione

Operazione	Description	Competenze richieste
	1. Tablespace 2. Profili 3. Roles Questo aiuta a garantire che la migrazione di Oracle Data Pump possa creare gli schemi e i relativi oggetti.	

Esegui una migrazione a pieno carico utilizzando Oracle Data Pump Import su un collegamento al database

Operazione	Description	Competenze richieste
Esegui la migrazione degli schemi richiesti.	Per migrare gli schemi richiesti dal database locale di origine all'istanza Amazon RDS di destinazione, usa il codice nella sezione Informazioni aggiuntive: • Per migrare un singolo schema, esegui Code 1 dalla sezione Informazioni aggiuntive. • Per migrare più schemi, esegui Code 2 dalla sezione Informazioni aggiuntive. Per ottimizzare le prestazioni della migrazione, puoi regolare il numero di processi paralleli	DBA

Operazione	Description	Competenze richieste
	eseguendo il comando seguente: `DBMS_DATAPUMP.SET_ PARALLEL (handle => v_hdnl, degree => 4);`	
Raccogli le statistiche dello schema per migliorare le prestazioni.	Il comando Gather Schema Statistics restituisce le statistiche di Oracle Query Optimizer raccolte per gli oggetti del database. Utilizzando queste informazioni, l'ottimizzatore può selezionare il piano di esecuzione migliore per qualsiasi query relativa a questi oggetti: `EXECUTE DBMS_STATS.GATHER_SCHEMA_STATS(ownname => '<schema_name>');`	DBA

Esegui una migrazione a pieno carico e una replica CDC utilizzando Oracle Data Pump e AWS DMS

Operazione	Description	Competenze richieste
Acquisisci l'SCN sul database Oracle locale di origine.	Acquisisci il numero di modifica del sistema (SCN) sul database Oracle locale di origine. Utilizzerai l'SCN per l'importazione a pieno carico e come punto di partenza per la replica CDC.	DBA

Operazione	Description	Competenze richieste
	Per generare l'SCN corrente sul database di origine, esegui la seguente istruzione SQL: `SELECT current_scn FROM V$DATABASE;`	

Operazione	Description	Competenze richieste
Esegui la migrazione a pieno carico degli schemi.	Per migrare gli schemi richiesti (`FULL LOAD`) dal database locale di origine all'istanza Amazon RDS di destinazione, procedi come segue: • Per migrare un singolo schema, esegui Code 3 dalla sezione Informazioni aggiuntive. • Per migrare più schemi, esegui Code 4 dalla sezione Informazioni aggiuntive. Nel codice, sostituiscilo `<CURRENT_SCN_VALUE _IN_SOURCE_DATABAS E>` con l'SCN che hai acquisito dal database di origine: `DBMS_DATAPUMP.SET_ PARAMETER (handle => v_hdnl, name => 'FLASHBACK_SCN', value => <CURRENT_SCN_VALUE _IN_SOURCE_DATABAS E>);` Per ottimizzare le prestazioni della migrazione, puoi regolare il numero di processi paralleli:	DBA

Operazione	Description	Competenze richieste
	`DBMS_DATAPUMP.SET_PARALLEL (handle => v_hdnl, degree => 4);`	
Disabilita i trigger negli schemi migrati.	Prima di iniziare l'attività AWS DMS solo CDC, disattivatela negli schemi migrati`TRIGGERS`.	DBA
Raccogli le statistiche dello schema per migliorare le prestazioni.	Il comando Gather Schema Statistics restituisce le statistiche di Oracle Query Optimizer raccolte per gli oggetti del database: `EXECUTE DBMS_STATS.GATHER_SCHEMA_STATS(ownname => '<schema_name>');` Utilizzando queste informazioni, l'ottimizzatore può selezionare il piano di esecuzione migliore per qualsiasi query su questi oggetti.	DBA

Operazione	Description	Competenze richieste
AWS DMS Da utilizzare per eseguire una replica continua dall'origine alla destinazione.	Utilizzato AWS DMS per eseguire una replica continua dal database Oracle di origine all'istanza Amazon RDS for Oracle di destinazione. Per ulteriori informazioni, consulta Creazione di attività per l'utilizzo continuo della replica AWS DMS e il post di blog How to work with native CDC support in. AWS DMS	DBA, ingegnere addetto alla migrazione

Passare ad Amazon RDS for Oracle

Operazione	Description	Competenze richieste
Abilita Multi-AZ sull'istanza 48 ore prima del cutover.	Se si tratta di un'istanza di produzione, consigliamo di abilitare la distribuzione Multi-AZ sull'istanza Amazon RDS per offrire i vantaggi dell'alta disponibilità (HA) e del disaster recovery (DR).	DBA, ingegnere addetto alla migrazione
Interrompi l'attività AWS DMS solo CDC (se CDC era acceso).	1. Assicurati che la latenza di origine e la latenza di destinazione nelle CloudWatch metriche Amazon dell' AWS DMS attività mostrino 0 secondi. 2. Interrompi l'attività solo CDC. AWS DMS	DBA

Operazione	Description	Competenze richieste
Abilita i trigger.	Abilita `TRIGGERS` quello che hai disabilitato prima della creazione dell'attività CDC.	DBA

Risorse correlate

AWS

- Preparazione di un database di origine Oracle autogestito per CDC utilizzando AWS DMS
- Creazione di attività per la replica continua utilizzando AWS DMS
- Implementazioni Multi-AZ per un'elevata disponibilità
- Come utilizzare il supporto CDC nativo in AWS DMS (post del blog)

documentazione Oracle

- DBMS_DATAPUMP

Informazioni aggiuntive

Codice 1: solo migrazione a pieno carico, schema di applicazione singolo

```
DECLARE
    v_hdnl NUMBER;
BEGIN
    v_hdnl := DBMS_DATAPUMP.OPEN(operation => 'IMPORT', job_mode  => 'SCHEMA',
 remote_link => '<DB LINK Name to Source Database>', job_name  => null);
    DBMS_DATAPUMP.ADD_FILE( handle => v_hdnl, filename  => 'import_01.log',  directory
 => 'DATA_PUMP_DIR', filetype  => dbms_datapump.ku$_file_type_log_file);
    DBMS_DATAPUMP.METADATA_FILTER(v_hdnl,'SCHEMA_EXPR','IN (''<schema_name>'')');  --
 To migrate one selected schema
    DBMS_DATAPUMP.METADATA_FILTER (hdnl, 'EXCLUDE_PATH_EXPR','IN (''STATISTICS'')'); --
 To prevent gathering Statistics during the import
    DBMS_DATAPUMP.SET_PARALLEL (handle => v_hdnl, degree => 4);  -- Number of parallel
 processes performing export and import
    DBMS_DATAPUMP.START_JOB(v_hdnl);
END;
```

```
/
```

Codice 2: solo migrazione a pieno carico, schemi di applicazioni multiple

```
DECLARE
    v_hdnl NUMBER;
BEGIN
    v_hdnl := DBMS_DATAPUMP.OPEN(operation => 'IMPORT', job_mode  => 'SCHEMA',
 remote_link => '<DB LINK Name to Source Database>', job_name  => null);
    DBMS_DATAPUMP.ADD_FILE( handle => v_hdnl, filename  => 'import_01.log',  directory
 => 'DATA_PUMP_DIR', filetype  => dbms_datapump.ku$_file_type_log_file);
    DBMS_DATAPUMP.METADATA_FILTER (v_hdnl, 'SCHEMA_LIST',
 '''<SCHEMA_1>'',''<SCHEMA_2>'', ''<SCHEMA_3>'''); -- To migrate multiple schemas
    DBMS_DATAPUMP.METADATA_FILTER (v_hdnl, 'EXCLUDE_PATH_EXPR','IN (''STATISTICS'')');
 -- To prevent gathering Statistics during the import
    DBMS_DATAPUMP.SET_PARALLEL (handle => v_hdnl, degree => 4);  -- Number of parallel
 processes performing export and import
    DBMS_DATAPUMP.START_JOB(v_hdnl);
END;
/
```

Codice 3: migrazione a pieno carico prima dell'operazione solo CDC, schema a singola applicazione

```
DECLARE
    v_hdnl NUMBER;
BEGIN
    v_hdnl := DBMS_DATAPUMP.OPEN(operation => 'IMPORT', job_mode  => 'SCHEMA',
 remote_link => '<DB LINK Name to Source Database>', job_name  => null);
    DBMS_DATAPUMP.ADD_FILE( handle => v_hdnl, filename  => 'import_01.log',  directory
 => 'DATA_PUMP_DIR', filetype  => dbms_datapump.ku$_file_type_log_file);
    DBMS_DATAPUMP.METADATA_FILTER(v_hdnl,'SCHEMA_EXPR','IN (''<schema_name>'')');  --
 To migrate one selected schema
    DBMS_DATAPUMP.METADATA_FILTER (v_hdnl, 'EXCLUDE_PATH_EXPR','IN (''STATISTICS'')');
 -- To prevent gathering Statistics during the import
    DBMS_DATAPUMP.SET_PARAMETER (handle => v_hdnl, name => 'FLASHBACK_SCN', value =>
 <CURRENT_SCN_VALUE_IN_SOURCE_DATABASE>); -- SCN required for AWS DMS CDC only task.
    DBMS_DATAPUMP.SET_PARALLEL (handle => v_hdnl, degree => 4);  -- Number of parallel
 processes performing export and import
    DBMS_DATAPUMP.START_JOB(v_hdnl);
END;
/
```

Codice 4: migrazione a pieno carico prima dell'operazione solo CDC, schemi di applicazione multipli

```
DECLARE
    v_hdnl NUMBER;
BEGIN
    v_hdnl := DBMS_DATAPUMP.OPEN (operation => 'IMPORT', job_mode  => 'SCHEMA',
 remote_link => '<DB LINK Name to Source Database>', job_name  => null);
    DBMS_DATAPUMP.ADD_FILE (handle => v_hdnl, filename  => 'import_01.log',  directory
 => 'DATA_PUMP_DIR', filetype  => dbms_datapump.ku$_file_type_log_file);
    DBMS_DATAPUMP.METADATA_FILTER (v_hdnl, 'SCHEMA_LIST',
 '''<SCHEMA_1>'',''<SCHEMA_2>'', ''<SCHEMA_3>'''); -- To migrate multiple schemas
    DBMS_DATAPUMP.METADATA_FILTER (v_hdnl, 'EXCLUDE_PATH_EXPR','IN (''STATISTICS'')');
 -- To prevent gathering Statistics during the import
    DBMS_DATAPUMP.SET_PARAMETER (handle => v_hdnl, name => 'FLASHBACK_SCN', value =>
 <CURRENT_SCN_VALUE_IN_SOURCE_DATABASE>); -- SCN required for AWS DMS CDC only task.
    DBMS_DATAPUMP.SET_PARALLEL (handle => v_hdnl, degree => 4);  -- Number of parallel
 processes performing export and import
    DBMS_DATAPUMP.START_JOB(v_hdnl);
END;
/
```

Scenario in cui un approccio misto alla migrazione può funzionare meglio

In rari scenari in cui il database di origine contiene tabelle con milioni di righe e colonne LOBSEGMENT di dimensioni molto grandi, questo modello rallenterà la migrazione. Oracle esegue la migrazione LOBSEGMENTs tramite il collegamento di rete una alla volta. Estrae una singola riga (insieme ai dati della colonna LOB) dalla tabella di origine e inserisce la riga nella tabella di destinazione, ripetendo il processo fino alla migrazione di tutte le righe. Oracle Data Pump tramite il collegamento al database non supporta il caricamento in blocco o il caricamento diretto tramite percorso per. LOBSEGMENTs

In questa situazione, consigliamo quanto segue:

- Salta le tabelle identificate durante la migrazione di Oracle Data Pump aggiungendo il seguente filtro di metadati:

  ```
  dbms_datapump.metadata_filter(handle =>h1, name=>'NAME_EXPR', value => 'NOT IN
   (''TABLE_1'',''TABLE_2'')');
  ```

- Utilizza un AWS DMS task (migrazione a pieno carico, con replica CDC se richiesta) per migrare le tabelle identificate. AWS DMS estrarrà più righe dal database Oracle di origine e le inserirà in un batch nell'istanza Amazon RDS di destinazione, migliorando le prestazioni.

Esegui la migrazione PeopleSoft da Oracle ad Amazon RDS Custom

Gaurav Gupta, Amazon Web Services

Riepilogo

Oracle PeopleSoft è una soluzione ERP (Enterprise Resource Planning) per processi a livello aziendale. PeopleSoft ha un'architettura a tre livelli: client, applicazione e database. PeopleSoft può essere eseguito su Amazon Relational Database Service (Amazon RDS). Ora puoi eseguire anche PeopleSoft Amazon RDS Custom, che fornisce l'accesso al sistema operativo sottostante.

Amazon RDS Custom for Oracle è un servizio di database gestito per applicazioni legacy, personalizzate e confezionate che richiedono l'accesso al sistema operativo e all'ambiente di database sottostanti. Quando migri il tuo database Oracle su Amazon RDS Custom, Amazon Web Services (AWS) può gestire le attività di backup e l'alta disponibilità, mentre puoi concentrarti sulla manutenzione PeopleSoft dell'applicazione e delle funzionalità. Per i fattori chiave da considerare per una migrazione, consulta le strategie di migrazione del database Oracle in AWS Prescriptive Guidance.

Questo modello si concentra sui passaggi per migrare un PeopleSoft database da Amazon Elastic Compute Cloud (Amazon EC2) ad Amazon RDS Custom utilizzando un backup Oracle Recovery Manager (RMAN). Utilizza un file system condiviso Amazon Elastic File System (Amazon EFS) tra l'istanza EC2 e Amazon RDS Custom, sebbene sia possibile utilizzare anche Amazon FSx o qualsiasi unità condivisa. Il modello utilizza un backup completo RMAN (a volte indicato come backup di livello 0).

Prerequisiti e limitazioni

Prerequisiti

- Un database sorgente Oracle versione 19C in esecuzione su Amazon EC2 con Oracle Linux 7, Oracle Linux 8, Red Hat Enterprise Linux (RHEL) 7 o RHEL 8. Negli esempi di questo modello, il nome del database di origine è`FSDM092`, ma questo non è un requisito.

> **Note**
>
> È inoltre possibile utilizzare questo modello con i database di origine Oracle locali. È necessario disporre della connettività di rete appropriata tra la rete locale e un cloud privato virtuale (VPC).

- Un'istanza PeopleSoft demo 9.2.
- Un unico livello di PeopleSoft applicazione. Tuttavia, è possibile adattare questo modello per lavorare con più livelli di applicazione.
- Amazon RDS Custom configurato con almeno 8 GB di spazio di swap.

Limitazioni

Questo modello non supporta le seguenti configurazioni:

- Impostazione del `ARCHIVE_LAG_TARGET` parametro del database su un valore esterno all'intervallo 60-7200
- Disattivazione della modalità di registro dell'istanza DB () `NOARCHIVELOG`
- Disattivazione dell'attributo ottimizzato Amazon Elastic Block Store (Amazon EBS) dell'istanza EC2
- Modifica dei volumi EBS originali collegati all'istanza EC2
- Aggiungere nuovi volumi EBS o modificare il tipo di volume da gp2 a gp3
- Modifica del formato di estensione per il parametro (richiesto) `LOG_ARCHIVE_FORMAT *.arc`
- Multiplexazione o modifica della posizione e del nome del file di controllo (deve essere così) `/rdsdbdata/db/*DBNAME*/controlfile/control-01.ctl`

Per ulteriori informazioni su queste e altre configurazioni non supportate, consulta la documentazione di Amazon RDS.

Versioni del prodotto

Per le versioni e le classi di istanze di Oracle Database supportate da Amazon RDS Custom, consulta Requisiti e limitazioni per Amazon RDS Custom for Oracle.

Architecture

Stack tecnologico Target

- Application Load Balancer
- Amazon EFS
- Amazon RDS Custom per Oracle
- AWS Secrets Manager

- Amazon Simple Storage Service (Amazon S3)

Architettura di destinazione

Il seguente diagramma di architettura rappresenta un PeopleSoft sistema in esecuzione in una singola zona di disponibilità su AWS. È possibile accedere al livello dell'applicazione tramite un Application Load Balancer. Sia l'applicazione che i database si trovano in sottoreti private e l'istanza di database Amazon RDS Custom e Amazon EC2 utilizzano un file system condiviso Amazon EFS per archiviare e accedere ai file di backup RMAN. Amazon S3 viene utilizzato per creare il motore Oracle RDS personalizzato e per archiviare i metadati dei redo logs.

Tools (Strumenti)

Strumenti

Servizi AWS

- Amazon RDS Custom for Oracle è un servizio di database gestito per applicazioni legacy, personalizzate e confezionate che richiedono l'accesso al sistema operativo e all'ambiente di database sottostanti. Automatizza le attività di amministrazione del database, come backup e alta disponibilità.
- Amazon Elastic File System (Amazon EFS) ti aiuta a creare e configurare file system condivisi nel cloud AWS. Questo modello utilizza un file system condiviso Amazon EFS per archiviare e accedere ai file di backup RMAN.
- AWS Secrets Manager ti aiuta a sostituire le credenziali codificate nel codice, comprese le password, con una chiamata API a Secrets Manager per recuperare il segreto a livello di codice. In questo modello, si recuperano le password degli utenti del database da Secrets Manager per creare gli `ADMIN` utenti `RDSADMIN` e modificare le password `sys` `esystem`.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Elastic Load Balancing (ELB) distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni. Ad esempio, puoi distribuire il traffico tra istanze Amazon Elastic Compute Cloud (Amazon EC2), contenitori e indirizzi IP in una o più zone di disponibilità. Questo modello utilizza un Application Load Balancer.

Altri strumenti

- Oracle Recovery Manager (RMAN) fornisce supporto per il backup e il ripristino per i database Oracle. Questo modello utilizza RMAN per eseguire un backup a caldo del database Oracle di origine su Amazon EC2 che viene ripristinato su Amazon RDS Custom.

Best practice

- Per i parametri di inizializzazione del database, personalizza il pfile standard fornito dall'istanza database personalizzata di Amazon RDS PeopleSoft anziché utilizzare lo spfile dal database di origine Oracle. Questo perché gli spazi bianchi e i commenti causano problemi durante la creazione di repliche di lettura in Amazon RDS Custom. Per ulteriori informazioni sui parametri di inizializzazione del database, vedere la nota di supporto Oracle 1100831.1 (richiede un account Oracle Support).
- Amazon RDS Custom utilizza la gestione automatica della memoria Oracle per impostazione predefinita. Se desideri utilizzare il kernel Hugemem, puoi configurare Amazon RDS Custom per utilizzare invece la gestione automatica della memoria condivisa.
- Lascia il parametro abilitato per impostazione `memory_max_target` predefinita. Il framework lo utilizza in background per creare repliche di lettura.
- Abilita Oracle Flashback Database. Questa funzionalità è utile per ripristinare lo standby in scenari di test di failover (non di switchover).

Epiche

Configura l'istanza DB e il file system

Operazione	Description	Competenze richieste
Crea l'istanza DB.	Nella console Amazon RDS, crea un'istanza Amazon RDS Custom for Oracle DB con un nome DB chiamato FSDMO92 (o il nome del database di origine). Per istruzioni, consulta Working with Amazon RDS Custom nella documenta	DBA

Operazione	Description	Competenze richieste
	zione di AWS e il post di blog Amazon RDS Custom for Oracle — New Control Capabilities in Database Environment. Ciò garantisce che il nome del database sia impostato sullo stesso nome del database di origine. (Se lasciato vuoto, l'istanza EC2 e il nome del database verranno impostati su`ORCL`.)	

Esegui un backup completo RMAN del database Amazon EC2 di origine

Operazione	Description	Competenze richieste
Crea uno script di backup.	Crea uno script di backup RMAN per eseguire il backup del database sul file system Amazon EFS che hai montato (`/efs`nell'esempio seguente) . Puoi utilizzare il codice di esempio o eseguire uno degli script RMAN esistenti. `#!/bin/bash` `Dt=`date +'%Y%m%d-%H` `%M'` `BACKUP_LOG="rman-${O` `RACLE_SID}-$Dt"` `export TAGDATE=`date +` `%Y%m%d%H%M`;` `LOGPATH=/u01/scripts/l` `ogs`	DBA

Operazione	Description	Competenze richieste

	rman target / >> $LOGPATH/rman-${ORACLE_SID}-$Dt << EOF
	SQL "ALTER SYSTEM SWITCH LOGFILE";
	SQL "ALTER SESSION SET NLS_DATE_FORMAT="DD.MM.YYYY HH24:MI:SS"";
	RUN
	{
	ALLOCATE CHANNEL ch11 TYPE DISK MAXPIECESIZE 5G;
	ALLOCATE CHANNEL ch12 TYPE DISK MAXPIECESIZE 5G;
	BACKUP AS COMPRESSED BACKUPSET FULL DATABASE FORMAT '/efs/rman_backup/FSCM/%d_%T_%s_%p_FULL' ;
	SQL "ALTER SYSTEM ARCHIVE LOG CURRENT";
	BACKUP FORMAT '/efs/rman_backup/FSCM/%d_%T_%s_%p_ARCHIVE' ARCHIVELOG ALL DELETE ALL INPUT ;
	BACKUP CURRENT CONTROLFILE FORMAT '/efs/rman_backup/FSCM/%d_%T_%s_%p_CONTROL';
	}
	EXIT;
	EOF	

Operazione	Description	Competenze richieste

Esegui lo script di backup.	Per eseguire lo script di backup RMAN, accedi come utente Oracle Home ed esegui lo script.
	`$  chmod a+x rman_backup.sh`
	`$ ./rman_backup.sh &`
		DBA

Operazione	Description	Competenze richieste

Verifica la presenza di errori e annota il nome del file di backup.	Verificate la presenza di errori nel file di registro RMAN. Se tutto sembra a posto, elenca il backup del file di controllo eseguendo il comando seguente.
	`RMAN> list backup of controlfile;`
	`using target database control file instead of recovery catalog`
	Annotate il nome del file di output.
	`List of Backup Sets ==========`
	`BS Key Type LV Size Device Type Elapsed Time Completion Time ------- ---- -- ---------- --------- -- ------------ ---------- 12 Full 21.58M DISK 00:00:01 13-JUL-22 BP Key: 12 Status: AVAILABLE Compressed: NO Tag: TAG20220713T150155 Piece Name: /efs/rman_backup/FSCM/FSDMO92_20220713_12_1_CONTROL`
		DBA

Operazione	Description	Competenze richieste
	`Control File Included: Ckp SCN: 165591599 85898 Ckp time: 13-JUL-22` Utilizzerai il file di controllo del backup `/efs/rman_backup/FSCM/FSDMO92_20220713_12_1_CONTROL` quando ripristini il database su Amazon RDS Custom.	

Chiudi il livello di applicazione di origine

Operazione	Description	Competenze richieste
Chiudi l'applicazione.	Per chiudere il livello di applicazione di origine, utilizzate l'`psadmin`utilità o l'utilità della riga di `psadmin` comando. 1. Per spegnere il server web, esegui il comando seguente. `psadmin -w shutdown -d "webserver domain name"` 2. Per spegnere il server delle applicazioni, esegui il comando seguente.	DBA, Amministratore PeopleSoft

Operazione	Description	Competenze richieste

	`psadmin -c shutdown -d "application server domain name"`
	3. Per chiudere lo scheduler dei processi, eseguite il comando seguente.
	`psadmin -p stop -d "process scheduler domain name"`	

Configurazione del database Amazon RDS Custom di destinazione

Operazione	Description	Competenze richieste
Installa il pacchetto rpm nfs-utils.	Per installare il `nfs-utils rpm` pacchetto, esegui il comando seguente. `$ yum install -y nfs-utils`	DBA
Monta lo storage EFS.	Ottieni il comando di montaggio di Amazon EFS dalla pagina della console Amazon EFS. Monta il file system EFS sull'istanza Amazon RDS utilizzando un client Network File System (NFS). `sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsize=1048`	DBA

Operazione	Description	Competenze richieste
	576,hard,timeo=600 ,retrans=2,noresvp ort fs-xxxxxxxxxx.efs. eu-west-1.amazonaw s.com:/ /efs sudo mount -t nfs4 -o nfsvers=4.1,rsize= 1048576,wsize=1048 576,hard,timeo=600 ,retrans=2,noresvp ort fs-xxxxxxxxxx.efs. eu-west-1.amazonaw s.com:/ /efs	

Rilascia il database iniziale e crea le directory in cui archiviare i file del database

Operazione	Description	Competenze richieste
Metti in pausa la modalità di automazione.	Devi mettere in pausa la modalità di automazione sulla tua istanza database personalizzata Amazon RDS prima di procedere con i passaggi successivi, per assicurarti che l'automazione non interferisca con l'attività di ripristino RMAN. Puoi mettere in pausa l'automazione utilizzando la console AWS o il comando AWS Command Line Interface (AWS CLI) (assicurati di aver prima configurato l'AWS CLI). aws rds modify-db- instance \	DBA

Operazione	Description	Competenze richieste
	`--db-instance-id entifier peoplesoft-fscm-92 \ --automation-mode all-paused \ --resume-full-automation-mode-minute 360 \ --region eu-west-1` Quando specifichi la durata della pausa, assicurati di lasciare abbastanza tempo per il ripristino RMAN. Questo dipende dalla dimensione del database di origine, quindi modificate il valore 360 di conseguenza. Inoltre, assicuratevi che il tempo totale dell'automazione sospesa non si sovrappon ga alla finestra di backup o di manutenzione del database.	

Operazione	Description	Competenze richieste

Crea e modifica il file dei parametri per PeopleSoft	Per creare e modificare il pfile per PeopleSoft, usa il pfile standard creato con l'istanza database personalizzata di Amazon RDS. Aggiungi i parametri necessari. PeopleSoft
	
	1. Passa a `rds user rdsdb` eseguendo il comando seguente.
	
	``
	$ sudo su - rdsdb
	``
	
	2. Accedere a SQL*Plus nel database di avvio e creare il pfile eseguendo il comando seguente.
	
	``
	SQL> create pfile
	 from spfile;
	``
	
	Questo crea il pfile in. `$ORACLE_HOME/dbs`
	3. Effettua un backup di questo file.
	4. Modifica il pfile per aggiungere o aggiornare PeopleSoft i parametri.
	
	``
	*._gby_hash_aggreg
	ation_enabled=false
	
	*._unnest_subqu
	ery=false
	``
		DBA

Operazione	Description	Competenze richieste

	`*.nls_language='AMERICAN'`
	
	`*.nls_length_semantics='CHAR'`
	
	`*.nls_territory='AMERICA'`
	
	`*.open_cursors=1000`
	
	`*.db_files=1200`
	
	`*.undo_tablespace='UNDOTBS1'`
	
	PeopleSoft i parametri correlati sono disponibili nella Oracle Support Note 1100831.1.
	
	5. Rimuove il riferimento spfile dal pfile.
	
	`*.spfile='/rdsdbin/oracle/dbs/spfileFSDM092.ora'`	

Operazione	Description	Competenze richieste

Elimina il database iniziale.	Per eliminare il database Amazon RDS Custom esistente, usa il codice seguente.
	`$ sqlplus / as sysdba`
	`SQL> shutdown immediate ;`
	`SQL> startup mount exclusive restrict;`
	`SQL> drop database;`
	`SQL> exit`	

Operazione	Description	Competenze richieste
Ripristina il database Amazon RDS Custom dal backup.	Ripristina il database utilizzando lo script seguente. Lo script ripristinerà prima il control file e poi ripristinerà l'intero database dalle parti di backup archiviate sul mount EFS. `#!/bin/bash` `Dt=`date +'%Y%m%d-%H%M'` `BACKUP_LOG="rman-${ORACLE_SID}-$Dt"` `export TAGDATE=`date +%Y%m%d%H%M`;` `LOGPATH=/rdsdbdata/scripts/logs` `rman target / >> $LOGPATH/rman-${ORACLE_SID}-$Dt << EOF` `restore controlfile from "/efs/rman_backup/FSCM/FSDMO92_20220713_12_1_CONTROL";` `alter database mount;` `run` `{` `set newname for database to '/rdsdbdata/db/FSDMO92_A/datafile/%f_%b';` `SET NEWNAME FOR TEMPFILE 1 TO '/rdsdbdata/db/FSDMO92_A/datafile/%f_%b';` `RESTORE DATABASE;` `SWITCH DATAFILE ALL;` `SWITCH TEMPFILE ALL;` `RECOVER DATABASE;` `}`	DBA

Operazione	Description	Competenze richieste
	EOF sqlplus / as sysdba >> $LOGPATH/rman-${OR ACLE_SID}-$Dt<<-EOF ALTER DATABASE RENAME FILE '/u01/psoft/db/ oradata/FSDMO92/redo0 1.log' TO '/rdsdbda ta/db/FSDMO92_A/on linelog/redo01.log'; ALTER DATABASE RENAME FILE '/u01/psoft/db/ oradata/FSDMO92/redo0 2.log' TO '/rdsdbda ta/db/FSDMO92_A/on linelog/redo02.log'; ALTER DATABASE RENAME FILE '/u01/psoft/db/ oradata/FSDMO92/redo0 3.log' TO '/rdsdbda ta/db/FSDMO92_A/on linelog/redo03.log'; alter database clear unarchived logfile group 1; alter database clear unarchived logfile group 2; alter database clear unarchived logfile group 3; alter database open resetlogs; EXIT EOF	

Recupera le password da Secrets Manager, crea utenti e modifica le password

Operazione	Description	Competenze richieste
Recupera la password da Secrets Manager.	Puoi eseguire questo passaggio utilizzando la console AWS o l'interfaccia a riga di comando AWS. I passaggi seguenti mostrano le istruzioni per la console. 1. Accedi alla Console di gestione AWS e apri la console Amazon RDS. 2. Nel riquadro di navigazione, scegli Database, quindi seleziona il database Amazon RDS. 3. Scegli la scheda Configurazione e annota l'ID della risorsa per l'istanza. Sarà nel formato `db-<ID>` (ad esempio,`db-73GJNHLGDNZNDOXNWXSECUW6LE`). 4. Apri la console Secrets Manager. 5. Scegli il segreto con lo stesso nome`do-not-delete-custom-<resource_id>` , dove `resource-id` si riferisce all'ID della risorsa che hai annotato nel passaggio 3.	DBA

Operazione	Description	Competenze richieste
	6. Scegli Retrieve secret value (Recupera il valore del segreto). Questa password sarà la stessa per gli `admin` utenti `sys` `systemrdsadmin`,, e.	

Operazione	Description	Competenze richieste

Creare l'utente RDSADMIN.	`RDSADMIN`è l'utente del database per il monitoraggio e l'orchestrazione dell'istanza database personalizzata di Amazon RDS. Poiché il database iniziale è stato eliminato e il database di destinazione è stato ripristinato dall'origine utilizzando RMAN, devi ricreare questo utente dopo l'operazione di ripristino per assicurarti che il monitoraggio di Amazon RDS Custom funzioni come previsto. È inoltre necessario creare un profilo e un tablespace separati per l'utente. `RDSADMIN`
	
	1. Immettere i seguenti comandi in un prompt SQL.
	
	`SQL> set echo on feedback on serverout on SQL> @?/rdbms/admin/utlpwdmg.sql SQL> ALTER PROFILE DEFAULT LIMIT FAILED_LOGIN_ ATTEMPTS UNLIMITED PASSWORD_LIFE_TIME UNLIMITED PASSWORD_VERIFY_F UNCTION NULL;`
		DBA

Operazione	Description	Competenze richieste
	2. Crea il profiloRDSADMIN. `SQL> set echo on feedback on serverout on SQL> alter session set "_oracle_script"=true; SQL> CREATE PROFILE RDSADMIN LIMIT COMPOSITE_LIMIT UNLIMITED SESSIONS_PER_USER UNLIMITED CPU_PER_SESSION UNLIMITED CPU_PER_CALL UNLIMITED LOGICAL_READS_PER_SESSION UNLIMITED LOGICAL_READS_PER_CALL UNLIMITED IDLE_TIME UNLIMITED CONNECT_TIME UNLIMITED PRIVATE_SGA UNLIMITED FAILED_LOGIN_ATTEMPTS 10 PASSWORD_LIFE_TIME UNLIMITED PASSWORD_REUSE_TIME UNLIMITED PASSWORD_REUSE_MAX UNLIMITED PASSWORD_VERIFY_FUNCTION NULL PASSWORD_LOCK_TIME 86400/86400`	

Operazione	Description	Competenze richieste
	PASSWORD_GRACE_TIME 604800/86400; 3. Crea il RDSADMIN tablespace. SQL> CREATE BIGFILE TABLESPACE rdsadmin '/rdsdbdata/db/FSD MO92_A/datafile/rd sadmin.dbf' DATAFILE SIZE 7M AUTOEXTEND ON NEXT 1m LOGGING ONLINE PERMANENT BLOCKSIZE 8192 EXTENT MANAGEMEN T LOCAL AUTOALLOCATE DEFAULT NOCOMPRES S SEGMENT SPACE MANAGEMENT AUTO; 4. Crea l'utente. RDSADMIN Sostituisci la RDSADMIN password con quella ottenuta in precedenza da Secrets Manager. SQL> CREATE USER rdsadmin IDENTIFIED BY xxxxxxxxxx DEFAULT TABLESPACE rdsadmin TEMPORARY TABLESPACE TEMP profile rdsadmin ; 5. Concedi i privilegi a RDSADMIN.	

Operazione	Description	Competenze richieste
	``` SQL> GRANT "CONNECT" TO RDSADMIN WITH ADMIN OPTION; SQL> GRANT "RESOURCE " TO RDSADMIN WITH ADMIN OPTION; SQL> GRANT "DBA" TO RDSADMIN; SQL> GRANT "SELECT_C ATALOG_ROLE" TO RDSADMIN WITH ADMIN OPTION; SQL> GRANT ALTER SYSTEM TO RDSADMIN; SQL> GRANT UNLIMITED TABLESPACE TO RDSADMIN; SQL> GRANT SELECT ANY TABLE TO RDSADMIN; SQL> GRANT ALTER DATABASE TO RDSADMIN; SQL> GRANT ADMINISTER DATABASE TRIGGER TO RDSADMIN; SQL> GRANT ANY OBJECT PRIVILEGE TO RDSADMIN WITH ADMIN OPTION; SQL> GRANT INHERIT ANY PRIVILEGES TO RDSADMIN; SQL> ALTER USER RDSADMIN DEFAULT ROLE ALL; ``` 6. Set the SYS, SYSTEM, and DBSNMP user profiles to RDSADMIN.	

Operazione	Description	Competenze richieste

	`SQL> set echo on feedback on serverout on`
	`SQL> alter user SYS profile RDSADMIN;`
	`SQL> alter user SYSTEM profile RDSADMIN;`
	`SQL> alter user DBSNMP profile RDSADMIN;`	

Crea l'utente principale.	Poiché il database iniziale è stato eliminato e il database di destinazione è stato ripristinato dall'origine utilizzando RMAN, è necessario ricreare l'utente principale. In questo esempio, il nome utente principale è. `admin`
	`SQL> create user admin identified by <password>;`
	`SQL> grant dba to admin`
		DBA

Operazione	Description	Competenze richieste

Cambia le password di sistema.	Modifica le password di sistema utilizzando la password recuperata da Secrets Manager.
	`SQL> alter user sys identified by xxxxxxxxxx;`
	`SQL> alter user system identified by xxxxxxxxxx;`
	Se non modifichi queste password, Amazon RDS Custom visualizza il messaggio di errore «L'utente di monitoraggio del database o le credenziali utente sono cambiate».
		DBA

Configura le voci TNS per Amazon RDS Custom e PeopleSoft

Operazione	Description	Competenze richieste
Configura il file tnsnames.	Per connetterti al database dal livello dell'applicazione, configura il `tnsnames.ora` file in modo da poterti connettere al database dal livello dell'applicazione. Nell'esempio seguente, è possibile notare che esiste un collegamento software al `tnsnames.ora` file, ma per	DBA

Operazione	Description	Competenze richieste

	impostazione predefinita il file è vuoto.
	
	`$ cd /rdsdbbin/oracle/network/admin`
	`$ ls -ltr`
	`-rw-r--r-- 1 rdsdb database 1536 Feb 14 2018 shrept.lst`
	`lrwxrwxrwx 1 rdsdb database 30 Apr 5 13:19 listener.ora -> /rdsdbdata/config/listener.ora`
	`lrwxrwxrwx 1 rdsdb database 28 Apr 5 13:19 sqlnet.ora -> /rdsdbdata/config/sqlnet.ora`
	`lrwxrwxrwx 1 rdsdb database 30 Apr 5 13:19 tnsnames.ora -> /rdsdbdata/config/tnsnames.ora`
	
	1. Create la `tsnames.ora` voce. A causa del modo in cui Amazon RDS automation analizza i file, devi assicurarti che la voce non contenga spazi bianchi, commenti o righe aggiuntive. In caso contrario, potresti riscontrare problemi durante l'utilizzo di alcuni di questi APIs, come create-db-instance-read -replica.	

Operazione	Description	Competenze richieste

	2. Sostituisci la porta, l'host e il SID in base ai requisiti del database PeopleSoft . Utilizzate il codice seguente come esempio.
	
	`$ vi tnsnames.ora`
	
	`FSDMO92=(DESCRIP TION = (ADDRESS_ LIST = (ADDRESS = (PROTOCOL = TCP)(HOST = x.x.x.x)(PORT = 1521))) (CONNECT_ DATA = (SERVER = DEDICATED) (SID = FSDMO92)))`
	
	3. Per confermare che il PeopleSoft database è raggiungibile, esegui il comando seguente.
	
	`$ tnsping FSDMO92`
	
	`TNS Ping Utility for Linux: Version 19.0.0.0.0 - Production on 14-JUL-2022 10:16:45`
	
	`Copyright (c) 1997, 2021, Oracle. All rights reserved.`
	
	`Used parameter files: /rdsdbbin/oracle/network/admin/sqlnet.ora`	

Operazione	Description	Competenze richieste
	`Used TNSNAMES adapter to resolve the alias Attempting to contact (DESCRIPTION = (ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCP)(HOST = x.x.x.x)(PORT = 1521))) (CONNECT_DATA = (SERVER = DEDICATED) (SID = FSDMO92))) OK (0 msec)`	

Crea il softlink spfile

Operazione	Description	Competenze richieste

Crea il softlink spfile.	1. Per creare spfile nella posizione `/rdsdbdata/admin/FSDMO92/pfile` , esegui il comando seguente.
	`SQL> create spfile='/rdsdbdata/admin/FSDMO92/pfile/spfileFSDMO92.ora' from pfile;`
	2. Passate allo `$ORACLE_HOME/dbs` spfile e create un collegamento software per lo spfile.
	`ln -s '/rdsdbdata/admin/FSDMO92/pfile/`
		DBA

Operazione	Description	Competenze richieste
	`spfileFSDMO92.ora' spfileFSDMO92.ora` 3. Dopo aver creato questo file, è possibile chiudere e avviare il database utilizzando lo spfile.	

Esegui le fasi successive alla migrazione

Operazione	Description	Competenze richieste
Convalida lo schema, le connessioni e le attività di manutenzione.	Per finalizzare la migrazione, esegui le seguenti attività. • Raccogli le statistiche dello schema. • Assicurati che il livello PeopleSoft dell'applicazione possa connettersi al nuovo database Amazon RDS Custom. • Configura i tuoi programmi di backup e manutenzione.	DBA

Risorse correlate

- Utilizzo di Amazon RDS Custom
- Amazon RDS Custom for Oracle — Nuove funzionalità di controllo nell'ambiente di database (post di blog)
- Integrazione di Amazon RDS Custom per Oracle con Amazon EFS (post di blog)
- Configurazione di Amazon RDS come PeopleSoft database Oracle (white paper AWS)

Esegui la migrazione della funzionalità Oracle ROWID a PostgreSQL su AWS

Rakesh Raghav e Ramesh Pathuri, Amazon Web Services

Riepilogo

Questo modello descrive le opzioni per la migrazione della funzionalità delle `ROWID` pseudocolonne in Oracle Database a un database PostgreSQL in Amazon Relational Database Service (Amazon RDS) per PostgreSQL, Amazon Aurora PostgreSQL Compatible Edition o Amazon Elastic Compute Cloud (Amazon). EC2

In un database Oracle, la pseudocolonna è l'indirizzo fisico di una riga in una tabella. `ROWID` Questa pseudocolonna viene utilizzata per identificare in modo univoco una riga anche se la chiave primaria non è presente in una tabella. PostgreSQL ha una pseudocolonna simile `ctid` chiamata, ma non può essere usata come. `ROWID` Come spiegato nella documentazione di PostgreSQL`ctid`, potrebbe cambiare se viene aggiornato o dopo ogni processo. `VACUUM`

Esistono tre modi per creare la funzionalità `ROWID` pseudocolonna in PostgreSQL:

- Usa una colonna chiave primaria invece di identificare una riga `ROWID` in una tabella.
- Utilizzate una primary/unique chiave logica (che potrebbe essere una chiave composita) nella tabella.
- Aggiungi una colonna con valori generati automaticamente e rendila una primary/unique chiave da `ROWID` imitare.

Questo modello illustra tutte e tre le implementazioni e descrive i vantaggi e gli svantaggi di ciascuna opzione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Esperienza di codifica procedurale Language/PostgreSQL (PL/pgSQL)
- Fonte Oracle Database
- Un cluster compatibile con Amazon RDS per PostgreSQL o Aurora PostgreSQL o un'istanza per ospitare il database PostgreSQL EC2

Limitazioni

- Questo modello fornisce `ROWID` soluzioni alternative per la funzionalità. PostgreSQL non fornisce un equivalente a in Oracle Database. `ROWID`

Versioni del prodotto

- PostgreSQL 11.9 o versione successiva

Architecture

Stack tecnologico di origine

- database Oracle

Stack tecnologico Target

- Compatibile con Aurora PostgreSQL, Amazon RDS per PostgreSQL o un'istanza con un database PostgreSQL EC2

Opzioni di implementazione

Esistono tre opzioni per ovviare alla mancanza di `ROWID` supporto in PostgreSQL, a seconda che la tabella abbia una chiave primaria o un indice univoco, una chiave primaria logica o un attributo di identità. La scelta dipende dalle tempistiche del progetto, dalla fase di migrazione corrente e dalle dipendenze dall'applicazione e dal codice del database.

Opzione	Description	Vantaggi	Svantaggi
Chiave primaria o indice univoco	Se la tabella Oracle ha una chiave primaria, puoi utilizzare gli attributi di questa chiave per identificare in modo univoco una riga.	• Nessuna dipendenza dalle funzionalità proprietarie del database. • Impatto minimo sulle prestazioni, poiché i campi	• Richiede modifiche al codice dell'applicazione e del database su cui si basa il passaggio `ROWID` ai campi della chiave primaria.

		chiave primari sono indicizzati.	
Chiave logica primary/ unique	Se la tabella Oracle ha una chiave primaria logica, è possibile utilizzare gli attributi di questa chiave per identific are in modo univoco una riga. Una chiave primaria logica è costituita da un attributo o da un insieme di attributi che possono identificare in modo univoco una riga, ma non viene applicata al database tramite un vincolo.	• Nessuna dipendenz a dalle funzional ità proprietarie del database.	• Richiede modifiche al codice dell'appl icazione e del database su cui si basa il passaggio `ROWID` ai campi chiave primari. • Impatto significa tivo sulle prestazio ni se gli attributi della chiave primaria logica non sono indicizzati. Tuttavia, è possibile aggiungere un indice univoco per evitare problemi di prestazioni.

Attributo di identità	se la tua tabella Oracle non ha una chiave primaria, puoi creare un campo aggiuntivo come`GENERATED ALWAYS AS IDENTITY`. Questo attributo genera un valore univoco ogni volta che i dati vengono inseriti nella tabella, quindi può essere utilizzato per identificare in modo univoco una riga per le operazioni DML (Data Manipulation Language).	• Nessuna dipendenza dalle funzionalità proprietarie del database. • Il database PostgreSQL popola l'attributo e ne mantiene l'unicità.	• Richiede modifiche al codice dell'applicazione e del database su `ROWID` cui si basa il passaggio all'attributo di identità. • Impatto significativo sulle prestazioni se il campo aggiuntivo non è indicizzato. Tuttavia, puoi aggiungere un indice per evitare problemi di prestazioni.

Tools (Strumenti)

- Amazon Relational Database Service (Amazon RDS) per PostgreSQL ti aiuta a configurare, gestire e scalare un database relazionale PostgreSQL nel cloud AWS.
- Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando. In questo modello, puoi utilizzare l'AWS CLI per eseguire comandi SQL tramite pgAdmin.
- pgAdmin è uno strumento di gestione open source per PostgreSQL. Fornisce un'interfaccia grafica che consente di creare, gestire e utilizzare oggetti di database.
- AWS Schema Conversion Tool (AWS SCT) supporta migrazioni di database eterogenei convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione.

Epiche

Identifica le tabelle di origine

Operazione	Description	Competenze richieste
Identifica le tabelle Oracle che utilizzano l'`ROWID`attributo.	Utilizza AWS Schema Conversion Tool (AWS SCT) per identificare le tabelle Oracle con `ROWID` funzionalità. Per ulteriori informazioni, consulta la documentazione di AWS SCT. oppure In Oracle, usa la `DBA_TAB_COLUMNS` vista per identificare le tabelle che hanno un `ROWID` attributo. Questi campi possono essere utilizzati per memorizzare caratteri alfanumerici da 10 byte. Determina l'utilizzo e convertili in un `VARCHAR` campo, se necessario.	DBA o sviluppatore
Identifica il codice che fa riferimento a queste tabelle.	Usa AWS SCT per generare un rapporto di valutazione della migrazione per identificare le procedure interessate da`ROWID`. Per ulteriori informazioni, consulta la documentazione di AWS SCT. oppure Nel database Oracle di origine, utilizza il campo di	DBA o sviluppatore

Operazione	Description	Competenze richieste
	testo della `dba_source` tabella per identificare gli oggetti che utilizzano `ROWID` funzionalità.	

Determina l'utilizzo della chiave primaria

Operazione	Description	Competenze richieste
Identifica le tabelle che non dispongono di chiavi primarie.	Nel database Oracle di origine, utilizzare `DBA_CONSTRAINTS` per identificare le tabelle che non dispongono di chiavi primarie. Queste informazioni ti aiuteranno a determinare la strategia per ogni tabella. Esempio: `select dt.* from dba_tables dt where not exists (select 1 from all_constraints ct where ct.owner = Dt.owner and ct.table_name = Dt.table_name and ct.constraint_type = 'P' ) and dt.owner = '{schema}'`	DBA o sviluppatore

Identifica e applica la soluzione

Operazione	Description	Competenze richieste
Applica le modifiche alle tabelle con una chiave primaria definita o logica.	Apportate le modifiche al codice dell'applicazione e del database mostrate nella sezione Informazioni aggiuntive per utilizzare una chiave primaria univoca o una chiave primaria logica per identificare una riga nella tabella.	DBA o sviluppatore
Aggiungi un campo aggiuntivo alle tabelle che non hanno una chiave primaria definita o logica.	Aggiungi un attributo di tipo`GENERATED ALWAYS AS IDENTITY`. Apporta le modifiche al codice dell'applicazione e del database mostrate nella sezione Informazioni aggiuntive.	DBA o sviluppatore
Aggiungi un indice se necessario.	Aggiungi un indice al campo aggiuntivo o alla chiave logica primaria per migliorare le prestazioni SQL.	DBA o sviluppatore

Risorse correlate

- PostgreSQL CTID (documentazione PostgreSQL)
- Colonne generate (documentazione PostgreSQL)
- Pseudocolonna ROWID (documentazione Oracle)

Informazioni aggiuntive

Le sezioni seguenti forniscono esempi di codice Oracle e PostgreSQL per illustrare i tre approcci.

Scenario 1: utilizzo di una chiave unica primaria

Negli esempi seguenti, si crea la tabella `testrowid_s1` con `emp_id` come chiave primaria.

Codice Oracle:

```
create table testrowid_s1 (emp_id integer, name varchar2(10), CONSTRAINT testrowid_pk
 PRIMARY KEY (emp_id));
INSERT INTO testrowid_s1(emp_id,name) values (1,'empname1');
INSERT INTO testrowid_s1(emp_id,name) values (2,'empname2');
INSERT INTO testrowid_s1(emp_id,name) values (3,'empname3');
INSERT INTO testrowid_s1(emp_id,name) values (4,'empname4');
commit;

SELECT rowid,emp_id,name FROM testrowid_s1;
ROWID                  EMP_ID NAME
------------------ ---------- ----------
AAAF3pAAAAAAAMOAAA          1 empname1
AAAF3pAAAAAAAMOAAB          2 empname2
AAAF3pAAAAAAAMOAAC          3 empname3
AAAF3pAAAAAAAMOAAD          4 empname4

UPDATE testrowid_s1 SET name = 'Ramesh' WHERE rowid = 'AAAF3pAAAAAAAMOAAB' ;
commit;

SELECT rowid,emp_id,name FROM testrowid_s1;
ROWID                  EMP_ID NAME
------------------ ---------- ----------
AAAF3pAAAAAAAMOAAA          1 empname1
AAAF3pAAAAAAAMOAAB          2 Ramesh
AAAF3pAAAAAAAMOAAC          3 empname3
AAAF3pAAAAAAAMOAAD          4 empname4
```

Codice PostgreSQL:

```
CREATE TABLE public.testrowid_s1
(
    emp_id integer,
    name character varying,
    primary key (emp_id)
);

insert into public.testrowid_s1 (emp_id,name) values
(1,'empname1'),(2,'empname2'),(3,'empname3'),(4,'empname4');
```

```
select emp_id,name from testrowid_s1;
 emp_id |   name
--------+----------
      1 | empname1
      2 | empname2
      3 | empname3
      4 | empname4

update testrowid_s1 set name = 'Ramesh' where emp_id = 2 ;

select emp_id,name from testrowid_s1;
 emp_id |   name
--------+----------
      1 | empname1
      3 | empname3
      4 | empname4
      2 | Ramesh
```

Scenario 2: utilizzo di una chiave logica primaria

Negli esempi seguenti, si crea la tabella `testrowid_s2` con `emp_id` come chiave primaria logica.

Codice Oracle:

```
create table testrowid_s2 (emp_id integer, name varchar2(10) );
INSERT INTO testrowid_s2(emp_id,name) values (1,'empname1');
INSERT INTO testrowid_s2(emp_id,name) values (2,'empname2');
INSERT INTO testrowid_s2(emp_id,name) values (3,'empname3');
INSERT INTO testrowid_s2(emp_id,name) values (4,'empname4');
commit;

SELECT rowid,emp_id,name FROM testrowid_s2;
ROWID                  EMP_ID NAME
------------------ ---------- ----------
AAAF3rAAAAAAAMeAAA          1 empname1
AAAF3rAAAAAAAMeAAB          2 empname2
AAAF3rAAAAAAAMeAAC          3 empname3
AAAF3rAAAAAAAMeAAD          4 empname4

UPDATE testrowid_s2 SET name = 'Ramesh' WHERE rowid = 'AAAF3rAAAAAAAMeAAB' ;
commit;

SELECT rowid,emp_id,name FROM testrowid_s2;
ROWID                  EMP_ID NAME
```

```
------------------ ---------- ----------
AAAF3rAAAAAAAMeAAA          1 empname1
AAAF3rAAAAAAAMeAAB          2 Ramesh
AAAF3rAAAAAAAMeAAC          3 empname3
AAAF3rAAAAAAAMeAAD          4 empname4
```

Codice PostgreSQL:

```
CREATE TABLE public.testrowid_s2
(
    emp_id integer,
    name character varying
);

insert into public.testrowid_s2 (emp_id,name) values
(1,'empname1'),(2,'empname2'),(3,'empname3'),(4,'empname4');

select emp_id,name from testrowid_s2;
 emp_id |   name
--------+----------
      1 | empname1
      2 | empname2
      3 | empname3
      4 | empname4

update testrowid_s2 set name = 'Ramesh' where emp_id = 2 ;

select emp_id,name from testrowid_s2;
 emp_id |   name
--------+----------
      1 | empname1
      3 | empname3
      4 | empname4
      2 | Ramesh
```

Scenario 3: utilizzo di un attributo di identità

Negli esempi seguenti, si crea la tabella `testrowid_s3` senza chiave primaria e utilizzando un attributo di identità.

Codice Oracle:

```
create table testrowid_s3 (name varchar2(10));
```

```
INSERT INTO testrowid_s3(name) values ('empname1');
INSERT INTO testrowid_s3(name) values ('empname2');
INSERT INTO testrowid_s3(name) values ('empname3');
INSERT INTO testrowid_s3(name) values ('empname4');
commit;

SELECT rowid,name FROM testrowid_s3;
ROWID              NAME
------------------ ----------
AAAF3sAAAAAAAMmAAA empname1
AAAF3sAAAAAAAMmAAB empname2
AAAF3sAAAAAAAMmAAC empname3
AAAF3sAAAAAAAMmAAD empname4

UPDATE testrowid_s3 SET name = 'Ramesh' WHERE rowid = 'AAAF3sAAAAAAAMmAAB' ;
commit;

SELECT rowid,name FROM testrowid_s3;
ROWID              NAME
------------------ ----------
AAAF3sAAAAAAAMmAAA empname1
AAAF3sAAAAAAAMmAAB Ramesh
AAAF3sAAAAAAAMmAAC empname3
AAAF3sAAAAAAAMmAAD empname4
```

Codice PostgreSQL:

```
CREATE TABLE public.testrowid_s3
(
    rowid_seq bigint generated always as identity,
    name character varying
);

insert into public.testrowid_s3 (name) values
('empname1'),('empname2'),('empname3'),('empname4');

select rowid_seq,name from testrowid_s3;
 rowid_seq |   name
-----------+----------
         1 | empname1
         2 | empname2
         3 | empname3
         4 | empname4
```

```
update testrowid_s3 set name = 'Ramesh' where rowid_seq = 2 ;

select rowid_seq,name from testrowid_s3;
 rowid_seq |   name
-----------+----------
         1 | empname1
         3 | empname3
         4 | empname4
         2 | Ramesh
```

Esegui la migrazione dei codici di errore del database Oracle a un database compatibile con Amazon Aurora PostgreSQL

Sai Parthasaradhi e Veeranjaneyulu Grandhi, Amazon Web Services

Riepilogo

Questo modello mostra come migrare i codici di errore di Oracle Database su un database Edition compatibile con Amazon Aurora PostgreSQL utilizzando una tabella di metadati predefinita.

I codici di errore del database Oracle non hanno sempre un codice di errore PostgreSQL corrispondente. Questa differenza nei codici di errore può rendere difficile la configurazione della logica di elaborazione delle procedure o delle funzioni nell'architettura PostgreSQL di destinazione.

È possibile semplificare il processo memorizzando i codici di errore del database di origine e di destinazione significativi per il PL/pgSQL programma in una tabella di metadati. Quindi, configura la tabella per contrassegnare i codici di errore del database Oracle validi e mapparli agli equivalenti PostgreSQL prima di continuare con la logica di processo rimanente. Se il codice di errore del database Oracle non è presente nella tabella dei metadati, il processo termina con l'eccezione. È quindi possibile esaminare manualmente i dettagli dell'errore e aggiungere il nuovo codice di errore alla tabella se il programma lo richiede.

Utilizzando questa configurazione, il database compatibile con Amazon Aurora PostgreSQL può gestire gli errori allo stesso modo del database Oracle di origine.

> **Note**
>
> La configurazione di un database PostgreSQL per gestire correttamente i codici di errore del database Oracle richiede in genere modifiche al database e al codice dell'applicazione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database Oracle di origine con servizi di istanza e listener attivi e funzionanti
- Un cluster compatibile con Amazon Aurora PostgreSQL che è attivo e funzionante

- Familiarità con Oracle Database
- Familiarità con i database PostgreSQL

Architecture

Il diagramma seguente mostra un esempio di flusso di lavoro di database compatibile con Amazon Aurora PostgreSQL per la convalida e la gestione dei codici di errore dei dati:

Il diagramma mostra il flusso di lavoro seguente:

1. Una tabella contiene i codici e le classificazioni di errore del database Oracle e i codici di errore e le classificazioni di errore PostgreSQL equivalenti. La tabella include una colonna valid_error che classifica se codici di errore specifici e predefiniti sono validi o meno.
2. Quando una PL/pgSQL funzione (func_processdata) genera un'eccezione, richiama una seconda funzione (error_validation). PL/pgSQL
3. La funzione error_validation accetta il codice di errore del database Oracle come argomento di input. Quindi, la funzione confronta il codice di errore in entrata con la tabella per verificare se l'errore è incluso nella tabella.
4. Se il codice di errore del database Oracle è incluso nella tabella, la funzione error_validation restituisce un valore TRUE e la logica del processo continua. Se il codice di errore non è incluso nella tabella, la funzione restituisce un valore FALSE e la logica del processo esce con un'eccezione.
5. Quando la funzione restituisce un valore FALSE, i dettagli dell'errore vengono esaminati manualmente dal responsabile funzionale dell'applicazione per determinarne la validità.
6. Il nuovo codice di errore viene quindi aggiunto manualmente alla tabella oppure no. Se il codice di errore è valido e aggiunto alla tabella, la funzione error_validation restituisce un valore TRUE la volta successiva che si verifica l'eccezione. Se il codice di errore non è valido e il processo deve fallire quando si verifica l'eccezione, il codice di errore non viene aggiunto alla tabella.

Stack tecnologico

- Amazon Aurora PostgreSQL
- pgAdmin
- Oracle SQL Developer

Tools (Strumenti)

- Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL.
- pgAdmin è uno strumento di amministrazione e sviluppo open source per PostgreSQL. Fornisce un'interfaccia grafica che semplifica la creazione, la manutenzione e l'uso degli oggetti del database.
- Oracle SQL Developer è un ambiente di sviluppo gratuito e integrato che semplifica lo sviluppo e la gestione di Oracle Database sia nelle implementazioni tradizionali che in quelle cloud.

Epiche

Esegui la migrazione dei codici di errore del database Oracle al tuo database compatibile con Amazon Aurora PostgreSQL

Operazione	Description	Competenze richieste

Crea una tabella nel database compatibile con Amazon Aurora PostgreSQL.	Esegui il seguente comando PostgreSQL CREATE TABLE:
	`(`
	`source_error_code numeric NOT NULL,`
	`target_error_code character varying NOT NULL,`
	`valid_error character varying(1) NOT NULL`
	`);`
		Sviluppatore PostgreSQL, Oracle, per PostgreSQL RDS/ Aurora

Aggiungere i codici di errore PostgreSQL e i codici di errore	Esegui il comando PostgreSQL INSERT per aggiungere i valori del codice di errore	Sviluppatore PostgreSQL, Oracle, per PostgreSQL RDS/ Aurora

Operazione	Description	Competenze richieste

del database Oracle corrispondenti alla tabella.	richiesti alla tabella error_codes.
	
	I codici di errore PostgreSQL devono utilizzare il tipo di dati variabile del carattere (valore SQLSTATE). I codici di errore Oracle devono utilizzare il tipo di dati numerico (valore SQLCODE).
	
	Esempio di istruzioni Insert:
	
	`insert into error_codes values (-1817,'22007','Y');`
	`insert into error_codes values (-1816,'22007','Y');`
	`insert into error_codes values (-3114,'08006','N');`
	
	Note
	Se rilevi eccezioni Java Database Connectivity (JDBC) specifiche di Oracle, devi sostituirle con eccezioni generiche tra database o passare a eccezioni specifiche di PostgreSQL.	

Operazione	Description	Competenze richieste
Crea una funzione per convalidare i codici di errore PL/pgSQL .	Crea una PL/pgSQL funzione eseguendo il comando PostgreSQL CREATE FUNCTION. Assicurati che la funzione esegua le seguenti operazioni: • Accetta i codici di errore Oracle generati da un programma. • Verifica se i codici di errore sono presenti nella tabella error_codes. • Restituisce il valore TRUE o FALSE, a seconda che il codice di errore sia presente o meno nella tabella dei metadati.	Sviluppatore PostgreSQL, Oracle, per PostgreSQL RDS/ Aurora

Operazione	Description	Competenze richieste
Esamina manualmente i nuovi codici di errore man mano che vengono registrati dalla funzione. PL/pgSQL	Esamina manualmente i nuovi codici di errore. Se un nuovo codice di errore è valido per il tuo caso d'uso, aggiungilo alla tabella error_codes eseguendo il comando PostgreSQL INSERT. oppure Se un nuovo codice di errore non è valido per il tuo caso d'uso, non aggiungerlo alla tabella. La logica del processo continuerà a fallire e terminerà con un'eccezione quando si verifica l'errore.	Sviluppatore PostgreSQL, Oracle, per PostgreSQL RDS/ Aurora

Risorse correlate

Appendice A. Codici di errore PostgreSQL (documentazione PostgreSQL)

Messaggi di errore del database (documentazione di Oracle Database)

Esegui la migrazione di SAP ASE su Amazon Aurora, compatibile EC2 con PostgreSQL utilizzando AWS SCT e AWS DMS

Amit Kumar e Ankit Gupta, Amazon Web Services

Riepilogo

Questo modello descrive come migrare un database SAP Adaptive Server Enterprise (SAP ASE) ospitato su un'istanza Amazon Elastic Compute Cloud (Amazon) verso l'edizione compatibile con EC2 Amazon Aurora PostgreSQL utilizzando AWS Schema Conversion Tool (AWS SCT) e AWS Database Migration Service (AWS DMS). Il modello si concentra sia sulle conversioni DDL (Data Definition Language) per gli oggetti archiviati sia sulla migrazione dei dati.

La compatibilità con Aurora PostgreSQL supporta i carichi di lavoro OLTP (Online Transaction Processing). Questo servizio gestito fornisce configurazioni scalabili automaticamente su richiesta. Può avviare, spegnere, scalare o ridurre automaticamente il database in base alle esigenze dell'applicazione. È possibile eseguire il database nel cloud senza gestire alcuna istanza di database. La compatibilità con Aurora PostgreSQL offre un'opzione conveniente per carichi di lavoro poco frequenti, intermittenti o imprevedibili.

Il processo di migrazione consiste in due fasi principali:

- Conversione dello schema del database utilizzando AWS SCT
- Migrazione dei dati utilizzando AWS DMS

Istruzioni dettagliate per entrambe le fasi sono fornite nella sezione Epics. Per informazioni sulla risoluzione dei problemi specifici dell'utilizzo di AWS DMS con i database SAP ASE, consulta Risoluzione dei problemi con SAP ASE nella documentazione di AWS DMS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database SAP ASE di origine su un' EC2 istanza con server, database e servizi di listener attivi e funzionanti
- Un database di destinazione compatibile con Aurora PostgreSQL

Limitazioni

- Il numero di porta per le connessioni deve essere 5432.
- La funzionalità huge_pages è attiva per impostazione predefinita ma può essere modificata.
- Point-in-time la granularità del ripristino (PITR) è di 5 minuti.
- La replica tra regioni non è attualmente disponibile.
- La dimensione massima di archiviazione per un database Aurora è di 128 TiB.
- È possibile creare fino a 15 repliche di lettura.
- Il limite di dimensione della tabella è vincolato solo dalla dimensione del volume del cluster Aurora, quindi la dimensione massima della tabella per un cluster DB compatibile con Aurora PostgreSQL è di 32 TiB. Ti consigliamo di seguire le migliori pratiche per la progettazione delle tabelle, come il partizionamento di tabelle di grandi dimensioni.

Versioni del prodotto

- Database di origine: AWS DMS attualmente supporta SAP ASE 15, 15.5, 15.7 e 16.x. Consulta la AWS DMS User Guide per le informazioni più recenti sul supporto della versione SAP ASE.
- Database di destinazione: PostgreSQL 9.4 e versioni successive (per la versione 9.x), 10.x, 11.x, 12.x, 13.x e 14.x. Consulta la AWS DMS User Guide per le ultime versioni di PostgreSQL supportate.
- Amazon Aurora 1.x o versione successiva. Per le informazioni più recenti, consulta le versioni del motore e le versioni del motore compatibili con Aurora PostgreSQL nella documentazione di Aurora.

Architecture

Stack tecnologico di origine

- Database SAP ASE in esecuzione su Amazon EC2

Stack tecnologico Target

- Database Aurora compatibile con PostgreSQL

Architettura di migrazione

Tools (Strumenti)

- Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL.
- AWS Schema Conversion Tool (AWS SCT) supporta migrazioni di database eterogenei convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione.
- AWS DMS supporta diversi database di origine e destinazione. Per ulteriori informazioni, consulta Sources for Data Migration e Targets for Data Migration nella documentazione di AWS DMS. Per il supporto più completo della versione e delle funzionalità, ti consigliamo di utilizzare la versione più recente di AWS DMS.

Epiche

Configura l'ambiente

Operazione	Description	Competenze richieste
Configura l'accesso alla rete nell' EC2 istanza di origine.	Configura i gruppi di sicurezza nell' EC2 istanza che ospita il database SAP ASE di origine. Per istruzioni, consulta i gruppi EC2 di sicurezza Amazon per le istanze Linux nella EC2 documentazione di Amazon.	Amministratore di sistema
Crea il tuo cluster DB di destinazione compatibile con Aurora PostgreSQL.	Installa, configura e avvia un cluster compatibile con Aurora PostgreSQL per il tuo database di destinazione. Per ulteriori informazioni, consulta Creazione di un cluster Amazon Aurora DB	DBA

Operazione	Description	Competenze richieste
	nella documentazione di Aurora.	
Imposta l'autorizzazione per il cluster DB di destinazione.	Configura gruppi di sicurezza e firewall per il database di destinazione. Per istruzioni, consulta Creazione di un cluster Amazon Aurora DB nella documentazione di Aurora.	DBA, amministratore di sistema

Converti lo schema del tuo database con AWS SCT

Operazione	Description	Competenze richieste
Avvia AWS SCT.	Avvia AWS SCT seguendo le istruzioni nella documenta zione di AWS SCT. AWS SCT fornisce un'interf accia utente basata su progetti per convertire automatic amente lo schema del database di origine SAP ASE in un formato compatibile con l'istanza DB Aurora PostgreSQ L di destinazione.	DBA
Crea endpoint AWS SCT.	Crea endpoint per i database SAP ASE di origine e PostgreSQL di destinazione. Per istruzioni, consulta la documentazione di AWS SCT.	DBA

Operazione	Description	Competenze richieste
Crea un rapporto di valutazione.	Crea un rapporto di valutazione della migrazione del database per valutare la migrazione e rilevare eventuali oggetti e funzioni incompatibili. Per istruzioni, consulta la documentazione di AWS SCT.	DBA
Convertire lo schema.	Converti lo schema del database seguendo le istruzioni nella documentazione di AWS SCT.	DBA
Convalida gli oggetti del database.	Se AWS SCT non è in grado di convertire un oggetto di database, ne identificherà il nome e altri dettagli. È necessario convertire questi oggetti manualmente. Per identificare queste discrepanze, segui le istruzioni nel post del blog AWS Convalida gli oggetti del database dopo la migrazione da SAP ASE ad Amazon RDS for PostgreSQL o Amazon Aurora PostgreSQL.	DBA

Analizza la migrazione di AWS DMS

Operazione	Description	Competenze richieste
Convalida le versioni del database di origine e di destinazione.	Verifica la compatibilità delle versioni del database SAP ASE con AWS DMS. Per ulteriori informazioni, consulta Sources for AWS DMS e Targets for AWS DMS nella documentazione di AWS DMS.	DBA
Identifica i requisiti per il tipo e la capacità di archiviazione.	Scegli la capacità di archiviazione appropriata per il database di destinazione in base alle dimensioni del database di origine.	DBA, amministratore di sistema
Scegli il tipo di istanza, la capacità e altre caratteristiche dell'istanza di replica.	Scegli il tipo di istanza, la capacità, le funzionalità di archiviazione e le funzionalità di rete che soddisfano i tuoi requisiti. Per indicazioni, consulta Scelta dell'istanza di replica AWS DMS giusta per la migrazione nella documentazione di AWS DMS.	DBA, amministratore di sistema
Identifica i requisiti di sicurezza dell'accesso alla rete.	Identifica i requisiti di sicurezza dell'accesso alla rete per i database di origine e di destinazione. Segui le indicazioni in Configurazione di una rete	DBA, amministratore di sistema

Operazione	Description	Competenze richieste
	per un'istanza di replica nella documentazione di AWS DMS.	

Migrare i dati

Operazione	Description	Competenze richieste
Migra i dati creando un'attività di migrazione in AWS DMS.	Per migrare i dati, crea un'attività e segui le istruzioni nella documentazione di AWS DMS. Ti consigliamo di utilizzar e la versione più recente di AWS DMS per il supporto più completo della versione e delle funzionalità.	DBA
Convalida i dati.	Per verificare che i tuoi dati siano stati migrati con precisione dal database di origine al database di destinazione, segui le linee guida sulla convalida dei dati fornite nella documentazione di AWS DMS.	DBA

Migrare l'applicazione

Operazione	Description	Competenze richieste
Identifica la strategia di migrazione delle applicazioni.	Scegli una delle sette strategie (7R) per la migrazione delle applicazioni sul cloud.	DBA, proprietario dell'app, amministratore di sistema
Segui la strategia di migrazione delle applicazioni.	Completa le attività del database identificate dal team dell'applicazione, incluso l'aggiornamento dei dettagli di connessione DNS per il database di destinazione e l'aggiornamento delle query dinamiche.	DBA, proprietario dell'app, amministratore di sistema

Passa al database di destinazione

Operazione	Description	Competenze richieste
Trasferisci i client applicativi alla nuova infrastruttura.	Passa la connessione dal database di origine al database di destinazione. Per ulteriori informazioni, consulta la sezione Cut over della strategia di migrazione per i database relazionali.	DBA, proprietario dell'app, amministratore di sistema

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse AWS temporanee.	Termina tutte le attività di migrazione, le istanze di	DBA, amministratore di sistema

Operazione	Description	Competenze richieste
	replica, gli endpoint e altre risorse AWS SCT e AWS DMS. Per ulteriori informazioni, consulta la documentazione di AWS DMS.	
Rivedi e convalida i documenti del progetto.	Convalida tutti i passaggi della documentazione del progetto per assicurarti che tutte le attività siano state completate con successo.	DBA, proprietario dell'app, amministratore di sistema
Chiudi il progetto.	Chiudi il progetto di migrazion e e fornisci eventuali feedback.	DBA, proprietario dell'app, amministratore di sistema

Risorse correlate

Riferimenti

- Abilita connessioni crittografate per le istanze DB PostgreSQL in Amazon RDS (AWS Prescriptive Guidance)
- Trasporta i database PostgreSQL tra due istanze DB Amazon RDS utilizzando pg_transport (AWS Prescriptive Guidance)
- Prezzi di Amazon Aurora
- Best practice con Amazon Aurora PostgreSQL Compatible Edition (documentazione Amazon Aurora)
- Documentazione AWS SCT
- Documentazione AWS DMS
- Utilizzo di un database SAP ASE come origine per AWS DMS

Tutorial e video

- Guida introduttiva ad AWS Database Migration Service
- AWS Database Migration Service (video)

Migrazione dei certificati SSL di Windows su un Application Load Balancer utilizzando ACM

Chandra Sekhar Yaratha e Igor Kovalchuk, Amazon Web Services

Riepilogo

Il modello fornisce linee guida per l'utilizzo di AWS Certificate Manager (ACM) per migrare i certificati Secure Sockets Layer (SSL) esistenti da siti Web ospitati su server locali o istanze Amazon Elastic Compute Cloud (Amazon EC2) su Microsoft Internet Information Services (IIS). I certificati SSL possono quindi essere utilizzati con Elastic Load Balancing on AWS.

SSL protegge i dati, conferma la tua identità, migliora il posizionamento nei motori di ricerca, aiuta a soddisfare i requisiti del Payment Card Industry Data Security Standard (PCI DSS) e migliora la fiducia dei clienti. Gli sviluppatori e i team IT che gestiscono questi carichi di lavoro desiderano che le loro applicazioni e infrastrutture Web, inclusi il server IIS e Windows Server, rimangano conformi alle loro politiche di base.

Questo modello prevede l'esportazione manuale dei certificati SSL esistenti da Microsoft IIS, la loro conversione dal formato Personal Information Exchange (PFX) al formato Private Enhanced Mail (PEM) supportato da ACM e quindi l'importazione in ACM nel tuo account AWS. Descrive inoltre come creare un Application Load Balancer per l'applicazione e configurare Application Load Balancer per utilizzare i certificati importati. Le connessioni HTTPS vengono quindi terminate sull'Application Load Balancer e non è necessario un ulteriore sovraccarico di configurazione sul server Web. Per ulteriori informazioni, consulta Creare un listener HTTPS per l'Application Load Balancer.

I server Windows utilizzano file con estensione pfx o p12 per contenere il file della chiave pubblica (certificato SSL) e il relativo file di chiave privata univoco. L'Autorità di certificazione (CA) ti fornisce il tuo file di chiave pubblica. Utilizzate il server per generare il file di chiave privata associato in cui è stata creata la richiesta di firma del certificato (CSR).

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un cloud privato virtuale (VPC) su AWS con almeno una sottorete privata e una pubblica in ogni zona di disponibilità utilizzata dai tuoi obiettivi
- IIS versione 8.0 o successiva in esecuzione su Windows Server 2012 o versione successiva

- Un'applicazione Web in esecuzione su IIS
- Accesso dell'amministratore al server IIS

Architecture

Stack tecnologico di origine

- Implementazione del server Web IIS con SSL per garantire che i dati vengano trasmessi in modo sicuro in una connessione crittografata (HTTPS)

Architettura di origine

Stack tecnologico Target

- Certificati ACM nel tuo account AWS
- Un Application Load Balancer configurato per utilizzare certificati importati
- Istanze di Windows Server nelle sottoreti private

Architettura Target

Tools (Strumenti)

- AWS Certificate Manager (ACM) ti aiuta a creare, archiviare e rinnovare certificati e chiavi SSL/TLS X.509 pubblici e privati che proteggono i tuoi siti Web e le tue applicazioni AWS.
- Elastic Load Balancing (ELB) distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni. Ad esempio, puoi distribuire il traffico tra EC2 istanze, contenitori e indirizzi IP in una o più zone di disponibilità.

Best practice

- Applica i reindirizzamenti del traffico da HTTP a HTTPS.
- Configura correttamente i gruppi di sicurezza per il tuo Application Load Balancer per consentire il traffico in entrata solo verso porte specifiche.
- Avvia le tue EC2 istanze in diverse zone di disponibilità per garantire un'elevata disponibilità.

- Configura il dominio dell'applicazione in modo che punti al nome DNS dell'Application Load Balancer anziché al suo indirizzo IP.
- Assicurati che l'Application Load Balancer abbia configurato i controlli di integrità a livello di applicazione.
- Configura la soglia per i controlli sanitari.
- Usa Amazon CloudWatch per monitorare l'Application Load Balancer.

Epiche

Esporta un file.pfx

Operazione	Description	Competenze richieste
Esporta il file.pfx da Windows Server.	Per esportare il certificato SSL come file.pfx dal gestore IIS locale in Windows Server: 1. Scegliete Start, Administrative, Internet Information Services (IIS) Manager. 2. Selezionate il nome del server e, in Sicurezza, fate doppio clic su Certificati server. 3. Scegliete il certificato che desiderate esportare, quindi scegliete Esporta. 4. Nella casella Esporta certificato, scegli una posizione, un percorso e un nome per il tuo file.pfx. 5. Specificate e confermate una password per il file.pfx.	Amministratore di sistema

Operazione	Description	Competenze richieste
	Note È necessaria questa password quando si installa il file.pfx. 6. Scegli OK. Il file.pfx dovrebbe ora essere salvato nella posizione e nel percorso specificati.	

Convertire il certificato con codifica PFX in formato PEM

Operazione	Description	Competenze richieste
Scarica e installa il toolkit OpenSSL.	1. Scarica e installa Win32/Win64 OpenSSL dal sito Web di Shining Light Productions. 2. Aggiungi la posizione dei binari OpenSSL alla variabile di PATH sistema, in modo che i binari possano essere disponibili per l'uso da riga di comando.	Amministratore di sistema
Convertire il certificato con codifica PFX in formato PEM.	I seguenti passaggi convertono il file di certificato firmato e codificato in PFX in tre file in formato PEM:	Amministratore di sistema

Operazione	Description	Competenze richieste
	• `cert-file.pem` contiene il SSL/TLS certificato per la risorsa. • `privatekey.pem` contiene la chiave privata del certificato senza protezione tramite password. • `ca-chain.pem` contiene il certificato radice della CA. Per convertire il certificato codificato PFX: 1. Esegui Windows. PowerShell 2. Utilizzate il seguente comando per estrarre la chiave privata del certificato dal file PFX. Immettete la password del certificato quando richiesto. `openssl pkcs12 -in <filename>.pfx -nocerts -out withpw-privatekey.pem` Il comando genera un file di chiave privata con codifica PEM denominato. `privatekey.pem` Immettete una passphrase per proteggere il file della	

Operazione	Description	Competenze richieste

	chiave privata quando richiesto.
	3. Eseguite il comando seguente per rimuovere la passphrase. Quando richiesto, fornite la passphrase creata nel passaggio 2.
	`openssl rsa -in withpw-privatekey.pem -out privatekey.pem`
	Se il comando ha esito positivo, viene visualizzato il messaggio «scrittura della chiave RSA».
	4. Utilizzate il seguente comando per trasferire il certificato dal file PFX a un file PEM.
	`openssl pkcs12 -in <file_name>.pfx -clcerts -nokeys -out cert-file.pem`
	Questo crea un file di certificato con codifica PEM denominato. `cert-file.pem` Se il comando ha esito positivo, viene visualizzato il messaggio «MAC verificato OK».	

Operazione	Description	Competenze richieste
	5. Crea un file di catena CA dal file PFX. Il comando seguente crea un file di catena CA denominato `ca-chain.pem` . `openssl pkcs12 -in <file_name>.pfx -cacerts -nokeys -chain -out ca-chain.pem` Se il comando ha esito positivo, viene visualizzato il messaggio «MAC verificato OK».	

Importa un certificato in ACM

Operazione	Description	Competenze richieste
Preparati a importare il certificato.	Nella console ACM, scegli Importa un certificato.	Amministratore cloud
Fornisci l'ente del certificato.	Per Organismo del certificato, incolla il certificato con codifica PEM che desideri importare. Per ulteriori informazioni sui comandi e sui passaggi descritti in questa e in altre attività di questa epopea, consulta Importazione di un certificato nella documenta zione ACM.	Amministratore cloud

Operazione	Description	Competenze richieste
Fornisci la chiave privata del certificato.	Per Certificate private key (Chiave privata certificato), incolla la chiave privata non crittografata, con codifica PEM, corrispondente alla chiave pubblica del certificato.	Amministratore cloud
Fornisci la catena di certificati.	Per Certificate Chain, incolla la catena di certifica ti con codifica PEM, che è memorizzata nel file. `CertificateChain.pem`	Amministratore cloud
Importa il certificato.	Seleziona Review and import (Riconsulta e importa). Verifica che le informazioni sul certifica to siano corrette, quindi scegli Importa.	Amministratore cloud

Creazione di un Application Load Balancer

Operazione	Description	Competenze richieste
Crea e configura il sistema di bilanciamento del carico e i listener.	Segui le istruzioni nella documentazione di Elastic Load Balancing per configura re un gruppo target, registrare i target e creare un Application Load Balancer e un listener. Aggiungi un secondo listener (HTTPS) per la porta 443.	Amministratore cloud

risoluzione dei problemi

Problema	Soluzione
Windows PowerShell non riconosce il comando OpenSSL anche dopo averlo aggiunto al percorso di sistema.	Assicurati `$env:path` che includa la posizione dei binari OpenSSL. In caso contrario, esegui il seguente comando in: PowerShell `$env:path = $env:path + ";C:\OpenSSL-Win64\bin"`

Risorse correlate

Importazione di un certificato in ACM

- Console ACM
- Certificato e formato della chiave per l'importazione
- Importazione di un certificato
- Guida per l'utente di AWS Certificate Manager

Creazione di un Application Load Balancer

- Creare un Application Load Balancer
- Guida per l'utente di Application Load Balancer

Esegui la migrazione di una coda di messaggistica da Microsoft Azure Service Bus ad Amazon SQS

Nisha Gambhir, Amazon Web Services

Riepilogo

Questo modello descrive come migrare un'applicazione web.NET Framework o.NET Core o console dall'uso della piattaforma di messaggistica in coda Microsoft Azure Service Bus ad Amazon Simple Queue Service (Amazon SQS).

Le applicazioni utilizzano i servizi di messaggistica per inviare e ricevere dati da altre applicazioni. Questi servizi aiutano a creare microservizi, sistemi distribuiti e applicazioni serverless disaccoppiati e altamente scalabili nel cloud.

Le code dei bus di servizio di Azure fanno parte di un'infrastruttura di messaggistica di Azure più ampia che supporta l'accodamento e la messaggistica. publish/subscribe

Amazon SQS è un servizio di accodamento dei messaggi completamente gestito che consente di disaccoppiare e scalare microservizi, sistemi distribuiti e applicazioni serverless. Amazon SQS elimina la complessità e il sovraccarico associati alla gestione e al funzionamento del middleware orientato ai messaggi e consente agli sviluppatori di concentrarsi sulla differenziazione del lavoro. Con Amazon SQS, puoi inviare, archiviare e ricevere messaggi tra componenti software a qualsiasi volume, senza perdere messaggi o richiedere la disponibilità di altri servizi.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un'applicazione web o console DI.NET Framework o .NET Core che usa le code dei bus di servizio di Azure (codice di esempio allegato)

Versioni del prodotto

- .NET Framework 3.5 o versione successiva oppure .NET Core 1.0.1, 2.0.0 o versione successiva

Architecture

Stack tecnologico di origine

- Un'applicazione web.NET (Core o Framework) o console che usa una coda del bus di servizio di Azure per inviare messaggi

Stack tecnologico Target

- Amazon SQS

Tools (Strumenti)

Strumenti

- Microsoft Visual Studio

Codice

Per creare una policy di AWS Identity and Access management (IAM) per Amazon SQS:

1. Accedi alla Console di gestione AWS, quindi apri la console di IAM all'indirizzo https://console.aws.amazon.com/iam/.

2. Nel pannello di navigazione sulla sinistra, selezionare Policies (Policy) e fare clic su Create Policy (Crea policy).

3. Scegli la scheda JSON e incolla il seguente codice:

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "VisualEditor0",
         "Effect": "Allow",
         "Action": [
            "sqs:DeleteMessage",
            "sqs:GetQueueUrl",
            "sqs:ChangeMessageVisibility",
            "sqs:SendMessageBatch",
            "sqs:ReceiveMessage",
            "sqs:SendMessage",
            "sqs:GetQueueAttributes",
            "sqs:ListQueueTags",
            "sqs:ListDeadLetterSourceQueues",
```

```
            "sqs:DeleteMessageBatch",
            "sqs:PurgeQueue",
            "sqs:DeleteQueue",
            "sqs:CreateQueue",
            "sqs:ChangeMessageVisibilityBatch",
            "sqs:SetQueueAttributes"
         ],
         "Resource": "arn:aws:sqs:*:<AccountId>:*"
      },
      {
         "Sid": "VisualEditor1",
         "Effect": "Allow",
         "Action": "sqs:ListQueues",
         "Resource": "*"
      }
   ]
}
```

4. Scegli Review policy, digita un nome, quindi scegli Crea policy.

5. Collega la policy appena creata al tuo ruolo IAM esistente o crea un nuovo ruolo.

Epiche

Configurare Amazon SQS in AWS

Operazione	Description	Competenze richieste
Crea una policy IAM per Amazon SQS.	Crea la policy IAM che fornirà l'accesso ad Amazon SQS. Consulta la sezione Codice per un esempio di policy.	Ingegnere di sistema
Crea un profilo AWS.	Crea un nuovo profilo eseguendo AWS Tools for PowerShell command Set-AWSCredential. Questo comando memorizza la chiave di accesso e la chiave segreta nel file di credenziali predefini to sotto il nome del profilo	Ingegnere di sistema

Operazione	Description	Competenze richieste
	specificato. Collega la policy Amazon SQS che hai creato in precedenza a questo account. Conserva l'ID della chiave di accesso AWS e la chiave di accesso segreta. Questi saranno necessari nei passaggi successivi.	
Crea una coda SQS.	È possibile creare una coda standard o una coda FIFO (First In-First Out). Per istruzioni, consulta il link nella sezione Riferimenti.	Ingegnere di sistema

Modifica il codice dell'applicazione.NET

Operazione	Description	Competenze richieste
Installa AWS Toolkit per Visual Studio.	Questo toolkit è un'estensione per Microsoft Visual Studio e semplifica la creazione e la distribuzione di applicazioni.NET in AWS. Per istruzioni di installazione e utilizzo, consulta il link nella sezione Riferimenti.	Sviluppatore di applicazioni
Installa il pacchetto AWSSDK .SQS. NuGet	Puoi installare AWSSDK .SQS scegliendo «Manage NuGet Package» in Visual Studio o eseguendo il comando « AWSSDKInstall-Package .SQS».	Sviluppatore di applicazioni

Operazione	Description	Competenze richieste
Crea un AWSCredentials oggetto nella tua applicazione.NET.	L'applicazione di esempio nell'allegato mostra come creare un AWSCredentials oggetto Basic, che eredita da AWSCredentials. È possibile utilizzare l'ID della chiave di accesso e la chiave di accesso segreta utilizzati in precedenza oppure lasciare che l'oggetto li scelga dalla cartella.aws come parte del profilo utente in fase di esecuzione.	Sviluppatore di applicazioni
Crea un oggetto client SQS.	Crea un oggetto client SQS (AmazonSQSClient) per.NET Framework. Fa parte dello spazio dei nomi Amazon.sqs. Questo oggetto è richiesto al posto di IQueue Client, che fa parte di Microsoft.Azure.ServiceBus spazio dei nomi.	Sviluppatore di applicazioni
Chiama il SendMessageAsync metodo per inviare messaggi alla coda SQS.	Cambia il codice che invia il messaggio alla coda per utilizzare il. amazonSqsClient SendMessageAsync metodo. Per i dettagli, consultate l'esempio di codice allegato.	Sviluppatore di applicazioni

Operazione	Description	Competenze richieste
Chiama il ReceiveMessageAsync metodo per ricevere messaggi dalla coda SQS.	Cambia il codice che riceve il messaggio per utilizzare il. amazonSqsClient ReceiveMessageAsync metodo. Per i dettagli, consultate l'esempio di codice allegato.	Sviluppatore di applicazioni
Chiama il DeleteMessageAsync metodo per eliminare i messaggi dalla coda SQS.	Per eliminare i messaggi, modifica il codice dal QueueClient. CompleteAsync metodo per. amazonSqsClient DeleteMessageAsync metodo. Per i dettagli, consultate l'esempio di codice allegato.	Sviluppatore di applicazioni

Risorse correlate

- Guida per gli sviluppatori dell'SDK AWS per .NET
- Messaggistica tramite Amazon SQS
- Creazione e utilizzo di una coda Amazon SQS con l'SDK AWS per.NET
- Inviare un messaggio Amazon SQS
- Ricevi un messaggio da una coda Amazon SQS
- Eliminazione di un messaggio da una coda Amazon SQS
- AWS Toolkit for Visual Studio

Informazioni aggiuntive

Questo modello include due applicazioni di esempio (consulta la sezione allegati):

- AzureSbTestAppinclude codice che usa la coda del bus di servizio di Azure.
- AmazonSqsTestApputilizza Amazon SQS. Si tratta di un'applicazione console che utilizza.NET Core 2.2 e include esempi per l'invio e la ricezione di messaggi.

Note:

- QueueClient è un oggetto di IQueue Client, che fa parte di Microsoft.Azure. ServiceBus namespace (incluso in Microsoft.Azure). ServiceBus NuGet pacchetto).
- amazonSqsClient è un oggetto di AmazonSQSClient, che fa parte dello spazio dei nomi Amazon.sqs (incluso nel pacchetto.SQS). AWSSDK NuGet
- A seconda di dove è in esecuzione il codice, ad esempio se è in esecuzione EC2, il ruolo deve avere l'autorizzazione per scrivere nella coda SQS.

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Esegui la migrazione di un database Oracle JD Edwards EnterpriseOne su AWS utilizzando Oracle Data Pump e AWS DMS

Thanigaivel Thirumalai, Amazon Web Services

Riepilogo

Puoi migrare ed eseguire il tuo database JD Edwards su Amazon Relational EnterpriseOne Database Service (Amazon RDS). Quando migri il tuo database su Amazon RDS, AWS può occuparsi delle attività di backup e della configurazione ad alta disponibilità, in modo che tu possa concentrarti sulla manutenzione EnterpriseOne dell'applicazione e delle sue funzionalità. Per un elenco completo dei fattori chiave da considerare durante il processo di migrazione, consulta le strategie di migrazione del database Oracle in AWS Prescriptive Guidance.

Esistono diversi modi per migrare un EnterpriseOne database, tra cui:

- Utilizzo di Oracle Universal Batch Engine (UBE) R98403 per la creazione di schemi e tabelle e utilizzo di AWS Database Migration Service (AWS DMS) per la migrazione
- Utilizzo di strumenti nativi DB per la creazione di schemi e tabelle e utilizzo di AWS DMS per la migrazione
- Utilizzo di strumenti nativi DB per la migrazione di dati esistenti (a pieno carico) e utilizzo di AWS DMS per attività di change data capture (CDC)

Questo modello copre la terza opzione. Spiega come migrare i EnterpriseOne database locali su Amazon RDS for Oracle utilizzando Oracle Data Pump con AWS DMS e la sua funzionalità CDC.

Oracle JD Edwards EnterpriseOne è una soluzione ERP (Enterprise Resource Planning) per organizzazioni che producono, costruiscono, distribuiscono, forniscono assistenza o gestiscono prodotti o risorse fisiche. JD Edwards EnterpriseOne supporta vari hardware, sistemi operativi e piattaforme di database.

Quando si migrano applicazioni ERP critiche come JD Edwards EnterpriseOne, ridurre al minimo i tempi di inattività è fondamentale. AWS DMS riduce al minimo i tempi di inattività supportando sia la replica a pieno carico che quella continua dal database di origine al database di destinazione. AWS DMS fornisce anche monitoraggio e registrazione in tempo reale per la migrazione, che possono aiutarti a identificare e risolvere eventuali problemi che potrebbero causare interruzioni.

Quando replichi le modifiche con AWS DMS, devi specificare un orario o un numero di modifica del sistema (SCN) come punto di partenza per leggere le modifiche dai log del database. È fondamentale

mantenere questi log accessibili sul server per un determinato periodo di tempo (consigliamo 15 giorni) per garantire che AWS DMS abbia accesso a queste modifiche.

Prerequisiti e limitazioni

Prerequisiti

- Un database Amazon RDS for Oracle fornito nel tuo ambiente cloud AWS come database di destinazione. Per istruzioni, consulta la documentazione di Amazon RDS.
- Un EnterpriseOne database in esecuzione in locale o su un'istanza Amazon Elastic Compute Cloud (Amazon EC2) su AWS.

> **Note**
>
> Questo modello è progettato per la migrazione da locale ad AWS, ma è stato testato utilizzando un EnterpriseOne database su un'istanza EC2. Se prevedi di migrare dal tuo ambiente locale, devi configurare la connettività di rete appropriata.

- Dettagli dello schema. Identifica lo schema del database Oracle (ad esempio DV920) per cui intendi migrare. EnterpriseOne Prima di iniziare il processo di migrazione, raccogli i seguenti dettagli sullo schema:
  - Dimensioni dello schema
  - Il numero di oggetti per tipo di oggetto
  - Il numero di oggetti non validi

Limitazioni

- Devi creare tutti gli schemi che desideri sul database Amazon RDS for Oracle di destinazione: AWS DMS non li crea per te. (La sezione Epics descrive come utilizzare Data Pump per esportare e importare schemi.) Il nome dello schema deve già esistere per il database Oracle di destinazione. Le tabelle dello schema di origine vengono importate nell'utente o nello schema e AWS DMS utilizza l'account amministratore o di sistema per connettersi all'istanza di destinazione. Per migrare più schemi, puoi creare più attività di replica. Puoi anche migrare i dati verso schemi diversi su un'istanza di destinazione. A tale scopo, utilizza le regole di trasformazione dello schema sulle mappature delle tabelle AWS DMS.
- Questo modello è stato testato con un set di dati dimostrativo. Ti consigliamo di convalidare la compatibilità del set di dati e della personalizzazione.

- Questo modello utilizza un EnterpriseOne database in esecuzione su Microsoft Windows. Tuttavia, puoi utilizzare lo stesso processo con altri sistemi operativi supportati da AWS DMS.

Architecture

Il diagramma seguente mostra un sistema in esecuzione EnterpriseOne su un database Oracle come database di origine e un database Amazon RDS for Oracle come database di destinazione. I dati vengono esportati dal database Oracle di origine e importati nel database Amazon RDS for Oracle di destinazione utilizzando Oracle Data Pump e replicati per gli aggiornamenti CDC utilizzando AWS DMS.

1. Oracle Data Pump estrae i dati dal database di origine e i dati vengono inviati alla destinazione del database Amazon RDS for Oracle.
2. I dati CDC vengono inviati dal database di origine a un endpoint di origine in AWS DMS.
3. Dall'endpoint di origine, i dati vengono inviati all'istanza di replica AWS DMS, dove viene eseguita l'attività di replica.
4. Una volta completata l'attività di replica, i dati vengono inviati all'endpoint di destinazione in AWS DMS.
5. Dall'endpoint di destinazione, i dati vengono inviati all'istanza del database Amazon RDS for Oracle.

Tools (Strumenti)

Servizi AWS

- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali.
- Amazon Relational Database Service (Amazon RDS) per Oracle ti aiuta a configurare, gestire e scalare un database relazionale Oracle nel cloud AWS.

Altri servizi

- Oracle Data Pump ti aiuta a spostare dati e metadati da un database all'altro ad alta velocità.

Best practice

Migrating LOBs (Migrazione in corso)

Se il database di origine contiene oggetti binari di grandi dimensioni (LOBs) che devono essere migrati al database di destinazione, AWS DMS offre le seguenti opzioni:

- Modalità LOB completa: AWS DMS migra tutto LOBs dal database di origine a quello di destinazione indipendentemente dalle dimensioni. Sebbene la migrazione sia più lenta rispetto alle altre modalità, il vantaggio è che i dati non vengono troncati. Per prestazioni migliori, è possibile creare un'attività separata sulla nuova istanza di replica per migrare le tabelle con dimensioni superiori a pochi LOBs megabyte.
- Modalità LOB limitata: specifichi la dimensione massima dei dati delle colonne LOB, che consente ad AWS DMS di preallocare le risorse e applicarle in blocco. LOBs Se la dimensione delle colonne LOB supera la dimensione specificata nell'attività, AWS DMS tronca i dati e invia avvisi al file di registro di AWS DMS. È possibile migliorare le prestazioni utilizzando la modalità LOB limitata se la dimensione dei dati LOB rientra nella dimensione LOB limitata.
- Modalità LOB in linea: è possibile migrare LOBs senza troncare i dati o rallentare le prestazioni dell'attività eseguendo repliche sia di piccole che di grandi dimensioni. LOBs Innanzitutto, specificate un valore per il `InlineLobMaxSize` parametro, che è disponibile solo quando la modalità LOB completa è impostata su. `true` Il task AWS DMS trasferisce le piccole operazioni LOBs in linea, il che è più efficiente. Quindi, AWS DMS migra i file di grandi dimensioni LOBs eseguendo una ricerca dalla tabella di origine. Tuttavia, la modalità LOB in linea funziona solo durante la fase di pieno carico.

Generazione di valori di sequenza

Durante il processo CDC di AWS DMS, i numeri di sequenza incrementali non vengono replicati dal database di origine. Per evitare discrepanze nei valori di sequenza, devi generare il valore di sequenza più recente dall'origine per tutte le sequenze e applicarlo al database Amazon RDS for Oracle di destinazione.

AWS Secrets Manager

Per aiutarti a gestire le tue credenziali, ti consigliamo di seguire le istruzioni contenute nel post del blog Manage your AWS DMS endpoint Credentials with AWS Secrets Manager.

Prestazioni

- Istanze di replica – Per indicazioni sulla scelta della dimensione migliore dell'istanza, consulta Selezione della dimensione migliore per un'istanza di replica nella documentazione di AWS DMS.
- Opzioni di connettività – Per evitare problemi di latenza, ti consigliamo di scegliere l'opzione di connettività giusta. AWS Direct Connect fornisce il percorso più breve per accedere alle risorse AWS, poiché è una connessione dedicata tra i data center aziendali e AWS. Durante il transito, il traffico di rete rimane sulla rete globale AWS e non passa mai su Internet. In questo modo si riduce la possibilità di incorrere in rallentamenti o aumenti imprevisti della latenza rispetto all'utilizzo di una VPN o della rete Internet pubblica.
- Larghezza di banda di rete – Per ottimizzare le prestazioni, verificate che la velocità di trasmissione della rete sia elevata. Se utilizzi un tunnel VPN tra il database di origine locale e AWS DMS, assicurati che la larghezza di banda sia sufficiente per il tuo carico di lavoro.
- Parallelismo delle attività – È possibile accelerare la replica dei dati caricando più tabelle in parallelo durante il pieno carico. Questo modello utilizza gli endpoint RDBMS, quindi questa opzione si applica solo al processo di caricamento completo. Il parallelismo delle attività è controllato dal `MaxFullLoadSubTasks` parametro, che determina quante sottoattività a pieno carico vengono eseguite in parallelo. Per impostazione predefinita, questo parametro è impostato su 8, il che significa che otto tabelle (se selezionate nella mappatura delle tabelle) vengono caricate insieme durante la modalità completa. È possibile modificare questo parametro nella sezione delle impostazioni dell'attività a caricamento completo dello script JSON relativo all'attività.
- Parallelismo delle tabelle – AWS DMS consente inoltre di caricare un'unica tabella di grandi dimensioni utilizzando più thread paralleli. Ciò è particolarmente utile per le tabelle di origine Oracle che contengono miliardi di record oltre a più partizioni e sottopartizioni. Se la tabella di origine non è partizionata, puoi utilizzare i limiti delle colonne per i carichi paralleli.
- Suddividi i carichi – Quando dividi i carichi tra più attività o istanze AWS DMS, ricorda i limiti delle transazioni quando acquisisci le modifiche.

Epiche

Usa Oracle Data Pump per esportare lo schema EnterpriseOne

Operazione	Description	Competenze richieste
Genera l'SCN.	Quando il database di origine è attivo e utilizzato dall' EnterpriseOne applicazione, avvia l'esportazione dei	DBA

Operazione	Description	Competenze richieste

	dati con Oracle Data Pump. È innanzitutto necessari o generare un numero di modifica del sistema (SCN) dal database di origine sia per la coerenza dei dati durante l'esportazione con Oracle Data Pump sia come punto di partenza per CDC in AWS DMS.
	
	Per generare l'SCN corrente dal database di origine, utilizza la seguente istruzione SQL:
	
	`SQL> select current_scn from v$database;`
	
	`CURRENT_SCN`
	`-----------`
	`30009727`
	
	Salva l'SCN generato. Utilizzer ai l'SCN per esportare i dati e creare l'attività di replica AWS DMS.	

Operazione	Description	Competenze richieste

Crea il file dei parametri.	Per creare un file di parametri per l'esportazione dello schema, è possibile utilizzare il codice seguente.
	
	`directory=DMS_DATA_PUMP_DIR logfile=export_dms.log dumpfile=export_dms_data.dmp schemas=<schema name> flashback_scn=<SCN from previous command>`
	
	Note
	Potete anche definirne uno personalizzato `DATA_PUMP_DIR` utilizzando i seguenti comandi, in base alle vostre esigenze.
	
	`SQL> CREATE OR REPLACE DIRECTORY DMS_DATA_PUMP_DIR AS '<Directory for dump>'; Directory created.`
	
	`SQL> GRANT READ, WRITE ON DIRECTORY DMS_DATA_PUMP_DIR TO SYSTEM; Grant succeeded.`
		DBA

Operazione	Description	Competenze richieste

Esporta lo schema.	Per eseguire l'esportazione, utilizzare l'`expdp`utilità come segue:
	
	`C:\Users\Administrator>expdp system/********@<DB Name> PARFILE='<Path to PAR file create above>'`
	
	`Export: Release 19.0.0.0.0 - Production on *** *** ** **:**:** **** Version 19.3.0.0.0`
	
	`Copyright (c) 1982, 2019, Oracle and/or its affiliates. All rights reserved.`
	
	`Connected to: Oracle Database 19c Standard Edition 2 Release 19.0.0.0.0 - Production Starting "SYSTEM"."SYS_EXPORT_SCHEMA_02": system/**********@<DB Name>PARFILE='E:\exp_dms_datapump.par' Processing object type SCHEMA_EXPORT/TABLE/TABLE_DATA Processing object type SCHEMA_EXPORT/TABLE/INDEX/STATISTICS/INDEX_STATISTICS Processing object type SCHEMA_EXPORT/TABL`
		DBA

Operazione	Description	Competenze richieste
	E/STATISTICS/TABLE _STATISTICS Processing object type SCHEMA_EXPORT/STAT ISTICS/MARKER Processing object type SCHEMA_EXPORT/USER Processing object type SCHEMA_EXPORT/ROLE _GRANT Processing object type SCHEMA_EXPORT/DEFA ULT_ROLE Processing object type SCHEMA_EXPORT/TABL ESPACE_QUOTA Processing object type SCHEMA_EXPORT/PRE_ SCHEMA/PROCACT_SCHEMA Processing object type SCHEMA_EXPORT/TABLE/ TABLE Processing object type SCHEMA_EXPORT/TABL E/GRANT/OWNER_GRANT/ OBJECT_GRANT Processing object type SCHEMA_EXPORT/TABLE/ INDEX/INDEX Processing object type SCHEMA_EXPORT/TABLE/ CONSTRAINT/CONSTRAINT . . exported "<Schema Name>"."<Table Name>" 228.9 MB 496397 rows Master table "SYSTEM". "SYS_EXPORT_SCHEMA _02" successfully loaded/unloaded	

Operazione	Description	Competenze richieste
	``` ********** ********** ********** ********** **** Dump file set for SYSTEM.SYS_EXPORT_ SCHEMA_02 is: E:\DMSDUMP\EXPORT_ DMS_DATA.DMP Job "SYSTEM"."SYS_EXPO RT_SCHEMA_02" successfully completed at *** *** ** **:**:** **** elapsed 0 00:01:57 ```	

Utilizzare Oracle Data Pump per importare lo schema EnterpriseOne

Operazione	Description	Competenze richieste
Trasferisci il file di dump nell'istanza di destinazione.	Per trasferire i file utilizzando l'`DBMS_FILE_TRANSFER` utilità, è necessario creare un collegamento al database dal database di origine all'istanza Amazon RDS for Oracle. Dopo aver stabilito il collegamento, puoi utilizzare l'utilità per trasferire i file Data Pump direttamente all'istanza Amazon RDS. In alternativa, puoi trasferire i file Data Pump su Amazon Simple Storage Service (Amazon S3) e quindi importarli nell'istanza Amazon	DBA

Operazione	Description	Competenze richieste

	RDS for Oracle. Per ulteriori informazioni su questa opzione, consulta la sezione Informazioni aggiuntive.
	
	Per creare un link al database ORARDSDB che si connette all'utente master di Amazon RDS nell'istanza DB di destinazione, esegui i seguenti comandi sul database di origine:
	
	`sqlplus / as sysdba`
	
	`SQL*Plus: Release 19.0.0.0.0 on *** *** ** **:**:** **** Version 19.3.0.0.0`
	
	`Copyright (c) 1982, 2019, Oracle. All rights reserved.`
	
	`Connected to: Oracle Database 19c Standard Edition 2 Release 19.0.0.0.0 Version 19.3.0.0.0`
	
	`SQL> create database link orardsdb connect to admin identifie d by "******" using '(DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(HOST = orcl.******.us-east-1.rds.amazonaws.com)(PORT = 1521))(CONNECT_DATA`	

Operazione	Description	Competenze richieste

	`= (SERVER = DEDICATED ) (SERVICE_NAME = orcl)))';`
	
	`Database link created.`
	
	`SQL>`	

Prova il link al database.	Verifica il collegamento al database per assicurarti di poterti connettere al database di destinazione Amazon RDS for Oracle utilizzan `dosqlplus`.
	
	`SQL> select name from v$database@orardsdb;`
	
	`NAME`
	`---------`
	`ORCL`
		DBA

Operazione	Description	Competenze richieste

Trasferisci il file di dump nel database di destinazione.	Per copiare il file di dump nel database Amazon RDS for Oracle, puoi utilizzare la directory `DATA_PUMP_DIR` predefinita oppure creare una directory personalizzata utilizzando il codice seguente, che deve essere eseguito sull'istanza Amazon RDS di destinazione:
	`exec rdsadmin.rdsadmin_util.create_directory(p_directory_name => 'DMS_TARGET_PUMP_DIR');`
	`PL/SQL procedure successfully completed.`
	Lo script seguente copia un file di dump denominato `EXPORT_DMS_DATA.DMP` dall'istanza di origine in un database Amazon RDS for Oracle di destinazione utilizzando il collegamento al database denominato. `orardsdb` È necessario eseguire lo script sull'istanza del database di origine.
	`BEGIN DBMS_FILE_TRANSFER.PUT_FILE(`
		DBA

Operazione	Description	Competenze richieste

	`source_directory_object => 'DMS_DATA_PUMP_DIR', source_file_name => 'EXPORT_DMS_DATA.DMP', destination_directory_object => 'DMS_TARGET_PUMP_DIR', destination_file_name => 'EXPORT_DMS_DATA.DMP', destination_database => 'orardsdb'); END;`
	
	`PL/SQL procedure successfully completed.`	

Elenca il file di dump nel database di destinazione.	Una volta completata la PL/SQL procedura, puoi elencare il file di dump dei dati nel database Amazon RDS for Oracle utilizzando il codice seguente: `select * from table (rdsadmin.rds_file_util.listdir(p_directory => 'DMS_TARGET_PUMP_DIR'));`	DBA

Operazione	Description	Competenze richieste

Crea utenti specifici per JDE nell'istanza di destinazione.	Crea un profilo e un ruolo JD Edwards utilizzando questi comandi nell'istanza di destinazione:
	`SQL> CREATE PROFILE "JDEPROFILE" LIMIT IDLE_TIME 15;`
	`Profile created.`
	
	`SQL> CREATE ROLE "JDE_ROLE";`
	`Role created.`
	
	`SQL> CREATE ROLE "JDEADMIN";`
	`CREATE ROLE "JDEUSER";`
	`Role created.`
	`Role created.`
	
	Concedi le autorizzazioni richieste al ruolo:
	`SQL> GRANT CREATE ANY SEQUENCE TO JDE_ROLE;`
	`GRANT DROP ANY SEQUENCE TO JDE_ROLE;`
	`GRANT CREATE ANY TRIGGER TO JDE_ROLE;`
	`GRANT DROP ANY TRIGGER TO JDE_ROLE;`
		DBA, JDE CNC

Operazione	Description	Competenze richieste

Crea tablespace nell'istanza di destinazione.	Crea i tablespace richiesti nell'istanza di destinazione utilizzando i seguenti comandi per gli schemi coinvolti in questa migrazione:
	`SQL> CREATE TABLESPACE <Tablespace Name for Tables>;`
	`Tablespace created.`
	
	`SQL> CREATE TABLESPACE <Tablespace Name for Indexes>;`
	`Tablespace created.`
		DBA, JDE CNC

Operazione	Description	Competenze richieste
Avvia l'importazione sul database di destinazione.	Prima di iniziare il processo di importazione, configura ruoli, schemi e tablespace sul database Amazon RDS for Oracle di destinazione utilizzando il file di dump dei dati. Per eseguire l'importazione, accedi al database di destinazione con l'account utente principale di Amazon RDS e usa il nome della stringa di connessione nel `tnsnames.ora` file, che include Amazon RDS for Oracle Database. `tns-entry` Se necessario, puoi includere un'opzione di rimappatura per importare il file di dump dei dati in una tablespace diversa o con un nome di schema diverso. Per avviare l'importazione, utilizzate il codice seguente: `impdp admin@orardsdb directory=DMS_TARGET_PUMP_DIR logfile=import.log dumpfile=EXPORT_DMS_DATA.DMP` Per garantire una corretta importazione, controllate il file di log di importazione per eventuali errori ed esaminate	DBA

Operazione	Description	Competenze richieste
	i dettagli come il conteggio degli oggetti, il conteggio delle righe e gli oggetti non validi. Se sono presenti oggetti non validi, ricompilali. Inoltre, confrontate gli oggetti del database di origine e di destinazione per confermare che corrispondano.	

Esegui il provisioning di un'istanza di replica AWS DMS con gli endpoint di origine e di destinazione

Operazione	Description	Competenze richieste
Eseguire il download del modello .	Scarica il modello AWS CloudFormation DMS_Instance.yaml per effettuare il provisioning dell'istanza di replica AWS DMS e dei relativi endpoint di origine e destinazione.	Amministratore del cloud, DBA
Inizia la creazione dello stack.	1. Accedi alla Console di gestione AWS e apri la CloudFormation console AWS all'indirizzo https://console.aws.amazon.com/cloudformation. 2. Seleziona Crea stack. 3. In Specify template (Specifica il modello), scegliere Upload a template file (Carica un file modello). 4. Scegli Scegli file.	Amministratore cloud, DBA

Operazione	Description	Competenze richieste
	5. Scegli il `DMS_instance.yaml` file. 6. Scegli Next (Successivo).	

Operazione	Description	Competenze richieste
Specificare i parametri.	1. Per il nome dello stack, inserisci il nome dello stack. 2. Per i parametri dell'istanza AWS DMS, inserisci i seguenti parametri: • DMSInstanceTipo: scegli l'istanza richiesta per l'istanza di replica AWS DMS, in base alle tue esigenze aziendali. • DMSStorageDimensioni: inserisci la dimensione dello storage per l'istanza AWS DMS, in base alla dimensione della migrazione. 3. Per Source Oracle Database Configuration, inserisci i seguenti parametri: • SourceOracleEndpointID: il nome del server del database Oracle di origine • SourceOracleDatabaseName— Il nome del servizio di database di origine o l'ID di sessione (SID), a seconda dei casi • SourceOracleUserName— Il nome utente del database di origine	Amministratore cloud, DBA

Operazione	Description	Competenze richieste
	(l'impostazione predefinita è `system`) • SourceOracleDBPassword— La password del nome utente del database di origine • SourceOracleDBPort— La porta del database di origine 4. Per Target RDS for Oracle Database Configuration, immettere i seguenti parametri: • Target RDSOracle EndpointID: l'endpoint del database RDS di destinazione • Target RDSOracle DatabaseName: il nome del database RDS di destinazione • Target RSOracle UserName: il nome utente RDS di destinazione • Target RDSOracle DBPassword: la password RDS di destinazione • TargetOracleDBPort— La porta del database RDS di destinazione	

Operazione	Description	Competenze richieste
	5. Per la configurazione di VPC, sottorete e gruppi di sicurezza, immettere i seguenti parametri: • VPCID: il VPC per l'istanza di replica • VPCSecurityGroupId— Il gruppo di sicurezza VPC per l'istanza di replica • DMSSubnet1— La sottorete per la zona di disponibilità 1 • DMSSubnet2— La sottorete per la zona di disponibilità 2 6. Scegli Next (Successivo).	
Creare lo stack.	1. Nella pagina Configura le opzioni dello stack, per Tag, inserisci eventuali valori opzionali. 2. Scegli Next (Successivo). 3. Nella pagina Revisione, verifica i dettagli, quindi scegli Invia. Il provisioning dovrebbe essere completato in circa 5-10 minuti. È completo quando la pagina AWS CloudFormation Stacks mostra CREATE_COMPLETE.	Amministratore cloud, DBA

Operazione	Description	Competenze richieste
Configura gli endpoint.	1. Apri la console AWS DMS alla versione https://console.aws.amazon.com/dms/v2/. 2. Per la gestione delle risorse, scegli Istanze di replica, quindi esamina le istanze di replica. 3. Per la gestione delle risorse, scegli Endpoints, quindi esamina gli endpoint.	Amministratore cloud, DBA
Prova la connettività.	Dopo che gli endpoint di origine e di destinazione hanno mostrato lo stato come Attivo, verifica la connettività. Scegli Esegui test per ogni endpoint (origine e destinazione) per assicurarti che lo stato risulti positivo.	Amministratore cloud, DBA

Crea un'attività di replica AWS DMS per la replica in tempo reale

Operazione	Description	Competenze richieste
Creare l'attività di replica.	Crea l'attività di replica di AWS DMS utilizzando i seguenti passaggi: 1. Apri la console AWS DMS alla versione https://console.aws.amazon.com/dms/v2/.	Amministratore cloud, DBA

Operazione	Description	Competenze richieste
	2. Nel pannello di navigazione, in Migrate Data, scegli Database migration task. 3. Nella casella di configurazione dell'attività, per Identificatore dell'attività, inserisci l'identificatore dell'attività. 4. Per Istanza di replica, scegli l'istanza di replica DMS che hai creato. 5. Per l'endpoint del database di origine, scegli l'endpoint di origine. 6. Per l'endpoint del database Target, scegli il database Amazon RDS for Oracle di destinazione. 7. Per il tipo di migrazione, scegli Replica solo le modifiche ai dati. Se ricevi un messaggio che indica che è necessario attivare la registrazione supplementare, segui le istruzioni nella sezione Risoluzione dei problemi. 8. Nella casella Impostazioni attività, scegli Specificare il numero di sequenza di registro. 9. Per il numero di modifica del sistema, inserisci	

Operazione	Description	Competenze richieste
	l'SCN del database Oracle generato dal database Oracle di origine. 10. Scegli Abilita convalida. 11. Scegli Abilita CloudWatch registri. Attivando questa funzionalità, puoi convalidare i dati e i log di Amazon per esaminare CloudWatch i log delle istanze di replica AWS DMS. 12. In Regole di selezione, completa quanto segue: • Per Schema, scegli Inserisci uno schema. • Per il nome dello schema, inserisci il nome dello schema JDE (ad esempio: DV920). • Per Nome tabella, inserisci%. • Per Azione, scegli Includi. 13. Scegli Create task (Crea attività). Dopo aver creato l'attività, AWS DMS migra le modifiche in corso all'istanza del database Amazon RDS for Oracle dall'SCN fornito in modalità di avvio CDC. Puoi	

Operazione	Description	Competenze richieste
	anche verificare la migrazione esaminando i log. CloudWatch	
Ripetere l'operazione di replica.	Ripeti i passaggi precedent i per creare attività di replica per altri schemi JD Edwards che fanno parte della migrazione.	Amministratore cloud, DBA, amministratore JDE CNC

Convalida lo schema del database sul database Amazon RDS for Oracle di destinazione

Operazione	Description	Competenze richieste
Convalida il trasferimento dei dati.	Dopo l'avvio del task AWS DMS, puoi controllare la scheda Table statistics nella pagina Tasks per vedere le modifiche apportate ai dati. Puoi monitorare lo stato della replica in corso nella console nella pagina Attività di migrazione del database. Per ulteriori informazioni, consulta la convalida dei dati di AWS DMS.	Amministratore del cloud, DBA

Tagliare

Operazione	Description	Competenze richieste
Interrompi la replica.	Interrompere la procedura di replica e interrompere i servizi applicativi di origine.	Amministratore cloud, DBA
Avvia l'applicazione JD Edwards.	Avvia la presentazione JD Edwards di destinazione e l'applicazione a livello logico su AWS e indirizzala al database Amazon RDS for Oracle. Quando accedi all'appli cazione, dovresti notare che tutte le connessioni sono ora stabilite con il database Amazon RDS for Oracle.	Amministratore DBA, JDE CNC
Disattiva il database di origine.	Dopo aver confermato che non ci sono più connessioni, puoi disattivare il database di origine.	DBA

Risoluzione dei problemi

Problema	Soluzione

Viene visualizzato un messaggio di avviso per abilitare la registrazione supplementare nel database di origine per la replica in corso	Immettete questi comandi per abilitare la registrazione supplementare:
	
	`SQL> ALTER DATABASE ADD SUPPLEMENTAL LOG DATA (ALL) COLUMNS;`
	`SQL> ALTER DATABASE ADD SUPPLEMENTAL LOG DATA (PRIMARY KEY) COLUMNS;`

Problema	Soluzione

	`SQL> ALTER DATABASE ADD SUPPLEMENTAL LOG DATA (UNIQUE) COLUMNS;`
	`SQL> ALTER DATABASE ADD SUPPLEMENTAL LOG DATA (FOREIGN KEY) COLUMNS;`
	`SQL> ALTER DATABASE ADD SUPPLEMENTAL LOG DATA (PRIMARY KEY) COLUMNS;`
	`SQL> ALTER DATABASE ADD SUPPLEMENTAL LOG DATA (UNIQUE) COLUMNS;`

AWS DMS ha disattivato la registrazione supplementare.	La registrazione supplementare è disattivata per impostazione predefinita in AWS DMS. Per attivarlo per un endpoint Oracle di origine: 1. Accedi alla Console di gestione AWS e apri la console AWS DMS alla versione https://console.aws.amazon.com/dms/v2/. 2. Scegli Endpoint. 3. Seleziona l'endpoint di origine Oracle a cui desideri aggiungere il log supplementare. 4. Scegli Modifica. 5. Seleziona Avanzato, quindi per Attributi aggiuntivi di connessione aggiungi il seguente codice: `addSupplementalLogging=Y` 6. Scegli Modifica.

La registrazione supplementare non è abilitata a livello di CDB.	1. Immettere il comando:
	`SQL> alter session set container = CDB$ROOT;`
	`Session altered.`
	2. Ripeti i passaggi per abilitare la registrazione supplementare.

Problema	Soluzione
Viene visualizzato il messaggio di errore: «Test Endpoint failed: Application-Status: 1020912, Application-Message: non è supportato in ambiente Oracle PDB L'inizializzazione dell'endpoint non LogMiner è riuscita».	Se viene visualizzato questo messaggio di errore, è possibile utilizzare Binary Reader anziché. LogMiner In Impostazioni endpoint, aggiungi questa riga agli attributi di connessione aggiuntivi per il tuo database di origine: `useLogMinerReader=N;useBfile=Y;`

Risorse correlate

- Guida introduttiva ad AWS Database Migration Service
- Best practice per AWS Database Migration Service
- Migrazione dei database Oracle sul cloud AWS
- Riferimento al tipo di risorsa AWS Database Migration Service per AWS CloudFormation
- Gestisci le credenziali degli endpoint AWS DMS con AWS Secrets Manager
- Risoluzione dei problemi di migrazione in AWS Database Migration Service
- Le migliori pratiche per AWS Database Migration Service

Informazioni aggiuntive

Trasferimento di file tramite Amazon S3

Per trasferire i file su Amazon S3, puoi utilizzare l'AWS CLI o la console Amazon S3. Dopo aver trasferito i file su Amazon S3, puoi utilizzare l'istanza Amazon RDS for Oracle per importare i file Data Pump da Amazon S3.

Se scegli di trasferire il file di dump utilizzando l'integrazione con Amazon S3 come metodo alternativo, procedi nel seguente modo:

1. Crea un bucket S3.
2. Esporta i dati dal database di origine utilizzando Oracle Data Pump.
3. Carica i file Data Pump nel bucket S3.

4. Scarica i file Data Pump dal bucket S3 al database Amazon RDS for Oracle di destinazione.
5. Esegui l'importazione utilizzando i file Data Pump.

> **Note**
>
> Per trasferire file di dati di grandi dimensioni tra istanze S3 e RDS, ti consigliamo di utilizzare la funzionalità Amazon S3 Transfer Acceleration.

Esegui la migrazione di un PeopleSoft database Oracle su AWS utilizzando AWS DMS

sampath kathirvel, Amazon Web Services

Riepilogo

Oracle PeopleSoft è una soluzione ERP (Enterprise Resource Planning) per processi a livello aziendale. PeopleSoft ha un'architettura a tre livelli: client, applicazione e database. PeopleSoft può essere eseguito su Amazon Relational Database Service (Amazon RDS).

Se migri il tuo database Oracle su Amazon RDS, Amazon Web Services (AWS) può occuparsi delle attività di backup e dell'elevata disponibilità, lasciandoti libero di concentrarti sulla manutenzione PeopleSoft dell'applicazione e delle sue funzionalità. Per un elenco completo dei fattori chiave da considerare durante il processo di migrazione, consulta le strategie di migrazione del database Oracle in AWS Prescriptive Guidance.

Questo modello fornisce una soluzione per la migrazione dei database Oracle locali su Amazon RDS for Oracle utilizzando Oracle Data Pump con AWS Database Migration Service (AWS DMS) e la sua funzionalità di acquisizione dei dati di modifica (CDC).

Durante la migrazione di applicazioni ERP critiche come PeopleSoft Oracle, è fondamentale ridurre al minimo i tempi di inattività. AWS DMS riduce al minimo i tempi di inattività supportando sia la replica a pieno carico che quella continua dal database di origine al database di destinazione. AWS DMS fornisce anche il monitoraggio e la registrazione in tempo reale della migrazione, che possono aiutarti a identificare e risolvere eventuali problemi che potrebbero causare tempi di inattività.

Quando si replicano le modifiche con AWS DMS, è necessario specificare un orario o un numero di modifica del sistema (SCN) come punto di partenza per consentire ad AWS DMS di leggere le modifiche dai log del database. È fondamentale mantenere questi log accessibili sul server per un determinato periodo di tempo per garantire che AWS DMS abbia accesso a queste modifiche.

Prerequisiti e limitazioni

Prerequisiti

- Hai effettuato il provisioning del database Amazon RDS for Oracle nel tuo ambiente cloud AWS come database di destinazione.
- Un PeopleSoft database Oracle in esecuzione in locale o su Amazon Elastic Compute Cloud (Amazon EC2) nel cloud AWS.

> **Note**
>
> Questo modello è progettato per la migrazione da locale ad AWS, ma è stato testato utilizzando Oracle Database su un' EC2 istanza Amazon. Per la migrazione da locale, dovrai configurare la connettività di rete appropriata.

- Dettagli dello schema. Quando si esegue la migrazione di un' PeopleSoft applicazione Oracle ad Amazon RDS for Oracle, è necessario identificare lo schema di database Oracle (ad esempio`SYSADM`) da migrare. Prima di iniziare il processo di migrazione, raccogli i seguenti dettagli sullo schema:
  - Dimensione
  - Il numero di oggetti per tipo di oggetto
  - Il numero di oggetti non validi.

  Queste informazioni saranno utili nel processo di migrazione.

Limitazioni

- Questo scenario è stato testato solo con il database PeopleSoft DEMO. Non è stato testato con un set di dati di grandi dimensioni.

Architecture

Il diagramma seguente mostra un'istanza che esegue un database Oracle come database di origine e un database Amazon RDS for Oracle come database di destinazione. I dati vengono esportati e importati dal database Oracle di origine al database Amazon RDS for Oracle di destinazione utilizzando Oracle Data Pump e replicati per le modifiche CDC utilizzando AWS DMS.

1. La fase iniziale prevede l'estrazione dei dati dal database di origine utilizzando Oracle Data Pump, seguita dall'invio alla destinazione del database Amazon RDS for Oracle.
2. I dati vengono inviati dal database di origine a un endpoint di origine in AWS DMS.
3. Dall'endpoint di origine, i dati vengono inviati all'istanza di replica AWS DMS, dove viene eseguita l'attività di replica.
4. Una volta completata l'attività di replica, i dati vengono inviati all'endpoint di destinazione in AWS DMS.

5. Dall'endpoint di destinazione, i dati vengono inviati all'istanza del database Amazon RDS for Oracle.

Tools (Strumenti)

Servizi AWS

- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali.
- Amazon Relational Database Service (Amazon RDS) per Oracle ti aiuta a configurare, gestire e scalare un database relazionale Oracle nel cloud AWS.

Altri servizi

- Oracle Data Pump ti aiuta a spostare dati e metadati da un database all'altro a velocità elevate.

Best practice

Migrating LOBs (Migrazione in corso)

Se il database di origine contiene oggetti binari di grandi dimensioni (LOBs) che devono essere migrati al database di destinazione, AWS DMS offre le seguenti opzioni:

- Modalità LOB completa: AWS DMS migra tutto LOBs dal database di origine a quello di destinazione indipendentemente dalle dimensioni. Sebbene la migrazione sia più lenta, il vantaggio è che i dati non vengono troncati. Per prestazioni migliori, è possibile creare un'attività separata sulla nuova istanza di replica per migrare le tabelle con dimensioni superiori a pochi megabyte. LOBs
- Modalità LOB limitata: specifichi la dimensione massima dei dati delle colonne LOB, che consente ad AWS DMS di preallocare le risorse e applicarle in blocco. LOBs Se la dimensione delle colonne LOB supera la dimensione specificata nell'attività, AWS DMS tronca i dati e invia avvisi al file di registro di AWS DMS. È possibile migliorare le prestazioni utilizzando la modalità LOB limitata se la dimensione dei dati LOB rientra nella dimensione LOB limitata.
- Modalità LOB in linea: è possibile migrare LOBs senza troncare i dati o rallentare le prestazioni dell'attività eseguendo repliche sia di piccole che di grandi dimensioni. LOBs Innanzitutto, specificate un valore per il InlineLobMaxSize parametro, che è disponibile solo quando la modalità Full LOB è impostata su true. Il task AWS DMS trasferisce le piccole operazioni LOBs in linea,

il che è più efficiente. Quindi, AWS DMS migra i file di grandi dimensioni LOBs eseguendo una ricerca dalla tabella di origine. Tuttavia, la modalità LOB in linea funziona solo durante la fase di pieno carico.

Generazione di valori di sequenza

Tieni presente che durante il processo di acquisizione dei dati di modifica con AWS DMS, i numeri di sequenza incrementali non vengono replicati dal database di origine. Per evitare discrepanze nei valori di sequenza, devi generare il valore di sequenza più recente dall'origine per tutte le sequenze e applicarlo al database Amazon RDS for Oracle di destinazione.

Gestione delle credenziali

Per proteggere le tue risorse AWS, ti consigliamo di seguire le best practice per AWS Identity and Access Management (IAM).

Epiche

Esegui il provisioning di un'istanza di replica AWS DMS con gli endpoint di origine e di destinazione

Operazione	Description	Competenze richieste
Eseguire il download del modello .	Scarica il CloudFormation modello AWS DMS_Instance.yaml per effettuare il provisioning dell'istanza di replica AWS DMS e dei relativi endpoint di origine e destinazione.	Amministratore cloud, DBA
Inizia la creazione dello stack.	1. Nella Console di gestione AWS, scegli CloudFormation. 2. Seleziona Crea stack. 3. In Specify template (Specifica il modello), scegliere Upload a template file (Carica un file modello).	Amministratore cloud, DBA

Operazione	Description	Competenze richieste
	4. Scegli Scegli il file. 5. Scegli il `DMS_instance.yaml` file. 6. Scegli Next (Successivo).	

Operazione	Description	Competenze richieste
Specificare i parametri.	1. Per il nome dello stack, inserisci il nome dello stack. 2. In Parametri dell'istanza AWS DMS, inserisci i seguenti parametri: • DMSInstanceTipo: scegli l'istanza richiesta per l'istanza di replica AWS DMS, in base alle tue esigenze aziendali. • DMSStorageDimensioni: inserisci la dimensione dello storage per l'istanza AWS DMS, in base alla dimensione della migrazione. 3. In Source Oracle Database Configuration, inserisci i seguenti parametri: • SourceOracleEndpoi ntID: il nome del server del database Oracle di origine • SourceOracleDataba seName— Il nome del servizio di database di origine o l'ID di sessione (SID), a seconda dei casi • SourceOracleUserNa me— Il nome utente del database di origine (l'impostazione predefini ta è system)	Amministratore cloud, DBA

Operazione	Description	Competenze richieste
	• SourceOracleDBPassword— La password del nome utente del database di origine • SourceOracleDBPort— La porta del database di origine 4. In Target RDS for Oracle Database Configuration, immettere i seguenti parametri: • Target RDSOracle EndpointID: l'endpoint del database RDS di destinazione • Target RDSOracle DatabaseName: il nome del database RDS di destinazione • Target RSOracle UserName: il nome utente RDS di destinazione • Target RDSOracle DBPassword: la password RDS di destinazione • TargetOracleDBPort— La porta del database RDS di destinazione 5. In Configurazione VPC, sottorete e gruppo di	

Operazione	Description	Competenze richieste
	sicurezza, inserisci i seguenti parametri: • VPCID: il VPC per l'istanza di replica • VPCSecurityGroupId— Il gruppo di sicurezza VPC per l'istanza di replica • DMSSubnet1— La sottorete per la zona di disponibilità 1 • DMSSubnet2— La sottorete per la zona di disponibilità 2 6. Scegli Next (Successivo).	
Creare lo stack.	1. Nella pagina Configura le opzioni dello stack, per Tag, inserisci eventuali valori opzionali. 2. Scegli Next (Successivo). 3. Nella pagina Revisione, verifica i dettagli, quindi scegli Invia. Il provisioning dovrebbe essere completato in circa 5-10 minuti. È completo quando la pagina AWS CloudFormation Stacks mostra CREATE_COMPLETE.	Amministratore del cloud, DBA

Operazione	Description	Competenze richieste
Configura gli endpoint.	1. Dalla Console di gestione AWS, scegli Database Migration Services. 2. In Gestione delle risorse, scegli Istanze di replica. 3. In Gestione delle risorse, scegli Endpoint.	Amministratore cloud, DBA
Prova la connettività.	Dopo che gli endpoint di origine e di destinazione hanno mostrato lo stato Attivo, verifica la connettività. Scegli Esegui test per ogni endpoint (origine e destinazione) per assicurarti che lo stato risulti positivo.	Amministratore cloud, DBA

Esporta lo PeopleSoft schema dal database Oracle locale utilizzando Oracle Data Pump

Operazione	Description	Competenze richieste
Genera l'SCN.	Quando il database di origine è attivo e utilizzato dall'appl icazione, avvia l'esporta zione dei dati con Oracle Data Pump. È innanzitu tto necessario generare un numero di modifica del sistema (SCN) dal database di origine sia per la coerenza dei dati durante l'esportazione con Oracle Data Pump sia come punto di partenza per l'acquisi	DBA

Operazione	Description	Competenze richieste

	zione dei dati di modifica in AWS DMS.
	
	Per generare l'SCN corrente dal database di origine, inserisci la seguente istruzione SQL.
	
	`SQL> select name from v $database;`
	`SQL> select name from v $database;`
	`NAME`
	`---------`
	`PSFTDMO`
	`SQL> SELECT current_s cn FROM v$database;`
	`CURRENT_SCN`
	`-----------`
	`23792008`
	
	Salva l'SCN generato da utilizzare quando esporti i dati e per creare l'attività di replica AWS DMS.	

Operazione	Description	Competenze richieste

Crea il file dei parametri.	Per creare un file di parametri per l'esportazione dello schema, è possibile utilizzare il codice seguente.
	
	`$ cat exp_datapmp.par`
	`userid=system/*******`
	`directory=DATA_P`
	`UMP_DIR`
	`logfile=export_dms_`
	`sample_user.log`
	`dumpfile=export_dms_`
	`sample_data_%U.dmp`
	`schemas=SYSADM`
	`flashback_scn=237920`
	`08`
	
	Note
	Potete anche definirne uno personalizzato `DATA_PUMP_DIR` utilizzando i seguenti comandi, in base alle vostre esigenze.
	
	`SQL> CREATE OR REPLACE`
	` DIRECTORY DATA_PUMP`
	`_DIR AS '/opt/oracle/`
	`product/19c/dbhome_1/`
	`dmsdump/';`
	`Directory created.`
	`SQL> GRANT READ, WRITE`
	` ON DIRECTORY DATA_PUMP`
	`_DIR TO system;`
	`Grant succeeded.`
		DBA

Operazione	Description	Competenze richieste
	SQL> SQL> SELECT owner, directory_name, directory_path FROM dba_directories WHERE directory_name='DA TA_PUMP_DIR'; OWNER DIRECTORY_NAME DIRECTORY_PATH ---------- ---------- ---------- ---------- ---------- ---------- SYS DATA_PUMP_DIR /opt/ oracle/product/19c/dbh ome_1/dmsdump/	

Operazione	Description	Competenze richieste
Esporta lo schema.	Per eseguire l'esportazione, utilizzare l'`expdp`utilità. `$ expdp parfile=e xp_datapmp.par .... Transferring the dump file with DBMS_FILE _TRANSFER to Target: . . exported "SYSADM". "PS_XML_TEMPLT_LNG" 6.320 KB 0 rows . . exported "SYSADM". "PS_XML_TEMPLT_LNK" 6.328 KB 0 rows . . exported "SYSADM". "PS_XML_XLATDEF_LNG" 6.320 KB 0 rows . . exported "SYSADM". "PS_XML_XLATITM_LNG" 7.171 KB 0 rows . . exported "SYSADM". "PS_XPQRYRUNCNTL" 7.601 KB 0 rows . . exported "SYSADM". "PS_XPQRYRUNPARM" 7.210 KB 0 rows . . exported "SYSADM". "PS_YE_AMOUNTS" 9.351 KB 0 rows . . exported "SYSADM". "PS_YE_DATA" 16.58 KB 0 rows . . exported "SYSADM". "PS_YE_EE" 6.75 KB 0 rows . . exported "SYSADM". "PS_YE_W2CP_AMOUNTS" 9.414 KB 0 rows`	DBA

Operazione	Description	Competenze richieste
	`. . exported "SYSADM". "PS_YE_W2CP_DATA" 20.94 KB 0 rows . . exported "SYSADM". "PS_YE_W2C_AMOUNTS" 10.27 KB 0 rows . . exported "SYSADM". "PS_YE_W2C_DATA" 20.95 KB 0 rows . . exported "SYSADM". "PS_ZBD_JOBCODE_TBL" 14.60 KB 0 rows . . exported "SYSADM". "PTGRANTTBL" 5.468 KB 0 rows Master table "SYSTEM". "SYS_EXPORT_SCHEMA _01" successfully loaded/unloaded ** Dump file set for SYSTEM.SYS_EXPORT_ SCHEMA_01 is: /opt/oracle/pr oduct/19c/dbhome_1 /dmsdump/export_dm s_sample_data_01.dmp Job "SYSTEM"."SYS_EXPO RT_SCHEMA_01" successfully completed at Mon Dec 19 20:13:57 2022 elapsed 0 00:38:22`	

Importa lo PeopleSoft schema nel database Amazon RDS for Oracle utilizzando Oracle Data Pump

Operazione	Description	Competenze richieste
Trasferisci il file di dump nell'istanza di destinazione.	Per trasferire i file utilizzan doDBMS_FILE_TRANSFER	DBA

Operazione	Description	Competenze richieste

	, devi creare un collegamento al database dal database di origine all'istanza Amazon RDS for Oracle. Dopo aver stabilito il collegamento, puoi utilizzare l'utilità per trasferire i file Data Pump direttamente all'istanza RDS.
	
	In alternativa, puoi trasferire i file Data Pump su Amazon Simple Storage Service (Amazon S3) e quindi importarli nell'istanza Amazon RDS for Oracle. Per ulteriori informazioni su questa opzione, consulta la sezione Informazioni aggiuntive.
	
	Per creare un link al database ORARDSDB che si connette all'utente master di Amazon RDS nell'istanza DB di destinazione, esegui i seguenti comandi sul database di origine.
	
	`$sqlplus / as sysdba`
	`$ SQL> create database link orardsdb connect to admin identified by "*****" using '(DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(HOST = testpsft.*******.us-west-2.rds.amazonaws.com)(PORT =`	

Operazione	Description	Competenze richieste
	1521))(CONNECT_DATA = (SERVER = DEDICATED) (SERVICE_NAME = orcl)))'; Database link created.	
Prova il link al database.	Verifica il collegamento al database per assicurarti di poterti connettere utilizzando sqlplus al database di destinazione Amazon RDS for Oracle. SQL> SQL> select name from v$database@orardsdb; NAME --------- ORCL SQL>	DBA

Operazione	Description	Competenze richieste

Trasferisci il file di dump nel database di destinazione.	Per copiare il file di dump sul database Amazon RDS for Oracle, puoi utilizzare la directory `DATA_PUMP_DIR` predefinita o creare la tua directory utilizzando il codice seguente.
	`exec rdsadmin.rdsadmin_util.create_directory(p_directory_name => 'TARGET_PUMP_DIR') ;`
	Lo script seguente copia un file di dump denominato `export_dms_sample_data_01.dmp` dall'istanza di origine in un database Amazon RDS for Oracle di destinazione utilizzando il collegamento al database denominato. `orardsdb`
	`$ sqlplus / as sysdba SQL> BEGIN DBMS_FILE_TRANSFER .PUT_FILE( source_directory_object => 'DATA_PUMP_DIR', source_file_name => 'export_dms_sample_data_01.dmp',`
		DBA

Operazione	Description	Competenze richieste
	`destination_directory _object => 'TARGET_P UMP_DIR'', destination_file_name => 'export_dms_sample _data_01.dmp', destination_database => 'orardsdb' ); END; / PL/SQL procedure successfully completed .`	
Elenca il file di dump nel database di destinazione.	Una volta completata la PL/SQL procedura, puoi elencare il file di dump dei dati nel database Amazon RDS for Oracle utilizzando il codice seguente. `SQL> select * from table (rdsadmin.rds_file _util.listdir(p_di rectory => 'TARGET_P UMP_DIR'));`	DBA

Operazione	Description	Competenze richieste
Avvia l'importazione sul database di destinazione.	Prima di iniziare il processo di importazione, configura ruoli, schemi e tablespace sul database Amazon RDS for Oracle di destinazione utilizzando il file di dump dei dati. Per eseguire l'importazione, accedi al database di destinazione con l'account utente principale di Amazon RDS e usa il nome della stringa di connessione nel `tnsnames.ora` file, che include Amazon RDS for Oracle Database. `tns-entry` Se necessario, puoi includere un'opzione di rimappatura per importare il file di dump dei dati in una tablespace diversa o con un nome di schema diverso. Per avviare l'importazione, utilizzate il codice seguente. `impdp admin@orardsdb directory=TARGET_PUMP_DIR logfile=import.log dumpfile=export_dms_sample_data_01.dmp` Per garantire una corretta importazione, controllate il file di log di importazione per	DBA

Operazione	Description	Competenze richieste
	eventuali errori ed esaminate i dettagli come il conteggio degli oggetti, il conteggio delle righe e gli oggetti non validi. Se sono presenti oggetti non validi, ricompilali. Inoltre, confrontate gli oggetti del database di origine e di destinazione per confermare che corrispondano.	

Crea un'attività di replica AWS DMS utilizzando CDC per eseguire la replica in tempo reale

Operazione	Description	Competenze richieste
Creare l'attività di replica.	Crea l'attività di replica di AWS DMS utilizzando i seguenti passaggi: 1. Nella console AWS DMS, in Conversione e migrazione, scegli Database migration task. 2. In Configurazione dell'atti vità, per Task identifier, inserisci il tuo identificatore di task. 3. Per Istanza di replica, scegli l'istanza di replica DMS che hai creato. 4. Per l'endpoint del database di origine, scegli l'endpoint di origine.	Amministratore cloud, DBA

Operazione	Description	Competenze richieste
	5. Per l'endpoint del database Target, scegli il tuo database Amazon RDS for Oracle di destinazione. 6. Per il tipo di migrazione, scegli Replica solo le modifiche ai dati. Se ricevi un messaggio che indica che è necessario attivare la registrazione supplementare, segui le istruzioni nella sezione Informazioni aggiuntive. 7. In Impostazioni attività, seleziona Specificare il numero di sequenza di registro. 8. Per il numero di modifica del sistema, inserisci l'SCN del database Oracle generato dal database Oracle di origine. 9. Scegli Abilita convalida. 10. Scegli Abilita CloudWatch registri. Attivando questa funzionalità, puoi convalidare i dati e i log di Amazon per esaminare CloudWatch i log delle istanze di replica AWS DMS. 11. In Regole di selezione, completa quanto segue:	

Operazione	Description	Competenze richieste
	• Per Schema, scegli Inserisci uno schema. • Per il nome dello schema, inserisci SYSADM. • Per Nome tabella, immettere%. • Per Azione, scegli Includi. 12. In Regole di trasformazione, completa quanto segue: • Per Target, scegli Tabella. • Per Nome dello schema, scegli Inserisci uno schema. • Per Nome schema, inserisci SYSADM. • Per Azione, scegli Rinomina in. 13. Scegli Create task (Crea attività). Dopo aver creato l'attività, migra il CDC all'istanza del database Amazon RDS for Oracle dall'SCN fornito in modalità di avvio CDC. Puoi anche verificare esaminando i log. CloudWatch	

Convalida lo schema del database sul database Amazon RDS for Oracle di destinazione

Operazione	Description	Competenze richieste
Convalida il trasferimento dei dati.	Dopo l'avvio del task AWS DMS, puoi controllare la scheda Table statistics nella pagina Tasks per vedere le modifiche apportate ai dati. Puoi monitorare lo stato della replica in corso nella console nella pagina Attività di migrazione del database. Per ulteriori informazioni, consulta la convalida dei dati di AWS DMS.	Amministratore del cloud, DBA

Tagliare

Operazione	Description	Competenze richieste
Interrompi la replica.	Interrompere la procedura di replica e interrompere i servizi applicativi di origine.	Amministratore cloud, DBA
Avvia il livello PeopleSoft intermedio.	Avvia l'applicazione di livello PeopleSoft intermedio di destinazione in AWS e indirizzala al database Amazon RDS for Oracle recentemente migrato. Quando accedi all'applicazione, dovresti notare che tutte le connessioni alle	DBA, amministratore PeopleSoft

Operazione	Description	Competenze richieste
	app sono ora stabilite con il database Amazon RDS for Oracle.	
Disattiva il database di origine.	Dopo aver verificato che non vi sono più connessioni al database di origine, è possibile disattivarlo.	DBA

Risorse correlate

- Guida introduttiva ad AWS Database Migration Service
- Best practice per AWS Database Migration Service
- Migrazione dei database Oracle sul cloud AWS

Informazioni aggiuntive

Trasferimento di file tramite Amazon S3

Per trasferire i file su Amazon S3, puoi utilizzare l'AWS CLI o la console Amazon S3. Dopo aver trasferito i file su Amazon S3, puoi utilizzare l'istanza Amazon RDS for Oracle per importare i file Data Pump da Amazon S3.

Se scegli di trasferire il file di dump utilizzando l'integrazione con Amazon S3 come metodo alternativo, procedi nel seguente modo:

1. Crea un bucket S3.
2. Esporta i dati dal database di origine utilizzando Oracle Data Pump.
3. Carica i file Data Pump nel bucket S3.
4. Scarica i file Data Pump dal bucket S3 al database Amazon RDS for Oracle di destinazione.
5. Esegui l'importazione utilizzando i file Data Pump.

> **Note**
>
> Per trasferire file di dati di grandi dimensioni tra istanze S3 e RDS, si consiglia di utilizzare la funzionalità Amazon S3 Transfer Acceleration.

Attiva la registrazione supplementare

Se si riceve un messaggio di avviso per abilitare la registrazione supplementare nel database di origine per la replica continua, utilizzare la procedura seguente.

```
SQL> ALTER DATABASE ADD SUPPLEMENTAL LOG DATA (ALL) COLUMNS;
SQL> ALTER DATABASE ADD SUPPLEMENTAL LOG DATA (PRIMARY KEY) COLUMNS;
SQL> ALTER DATABASE ADD SUPPLEMENTAL LOG DATA (UNIQUE) COLUMNS;
SQL> ALTER DATABASE ADD SUPPLEMENTAL LOG DATA (FOREIGN KEY) COLUMNS;
SQL> ALTER DATABASE ADD SUPPLEMENTAL LOG DATA (PRIMARY KEY) COLUMNS
SQL> ALTER DATABASE ADD SUPPLEMENTAL LOG DATA (UNIQUE) COLUMNS;
```

Esegui la migrazione di un database MySQL locale su Amazon RDS for MySQL

Lorenzo Mota, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni per la migrazione di un database MySQL locale ad Amazon Relational Database Service (Amazon RDS) per MySQL. Il modello illustra l'uso di AWS Database Migration Service (AWS DMS) o di strumenti MySQL nativi come mysqldump per una migrazione completa del database. Questo modello è destinato principalmente agli architetti di soluzioni. DBAs Può essere utilizzato in progetti piccoli o grandi come procedura di test (consigliamo almeno un ciclo di test) o come procedura di migrazione finale.

Prerequisiti e limitazioni

Prerequisiti

- Un AWS account attivo
- Un database sorgente MySQL in un data center locale

Limitazioni

- Limite di dimensione del database: 64 TB

Versioni del prodotto

- Versioni MySQL 5.5, 5.6, 5.7, 8.0. Per l'elenco più recente delle versioni supportate, consulta MySQL su Amazon AWS RDS nella documentazione. Se lo stai utilizzando AWS DMS, vedi anche Utilizzo di un database compatibile con MySQL come destinazione per le versioni di AWS DMS MySQL attualmente supportate da. AWS DMS

Architecture

Stack di tecnologia di origine

- Un database MySQL locale

Stack tecnologico Target

- Un'istanza database Amazon RDS che esegue MySQL

Architettura Target

Il diagramma seguente mostra l'implementazione di destinazione di Amazon RDS for MySQL dopo la migrazione.

AWS architettura di migrazione dei dati

Utilizzando AWS DMS:

Il diagramma seguente mostra l'architettura di migrazione dei dati utilizzata AWS DMS per inviare modifiche complete e incrementali fino al cutover. La connessione di rete da locale a AWS dipende dalle esigenze dell'utente e non rientra nell'ambito di questo schema.

Utilizzo di strumenti MySQL nativi:

Il diagramma seguente mostra l'architettura di migrazione dei dati quando si utilizzano strumenti MySQL nativi. I file di dump di esportazione vengono copiati su Amazon Simple Storage Service (Amazon S3) e importati nel database Amazon RDS for MySQL prima del cutover. AWS La connessione di rete da locale a AWS dipende dalle tue esigenze e non rientra nell'ambito di questo schema.

Note:

- A seconda dei requisiti di downtime e delle dimensioni del database, l'utilizzo AWS DMS o di uno strumento CDC (Change Data Capture) riduce al minimo i tempi di cutover. AWS DMS può aiutare a ridurre al minimo il tempo necessario per raggiungere il nuovo obiettivo (in genere minuti). Una strategia offline con mysqldump può essere sufficiente se le dimensioni del database e la latenza di rete consentono una finestra breve. (Si consiglia di eseguire il test per ottenere un orario approssimativo.)
- Di solito una strategia CDC come quella AWS DMS richiede un monitoraggio e una complessità maggiori rispetto alle opzioni offline.

Tools (Strumenti)

- Servizi AWS: AWS Database Migration Service (AWS DMS) consente di migrare gli archivi di dati verso Cloud AWS o tra combinazioni di configurazioni cloud e locali. Per informazioni sui database di origine e destinazione MySQL supportati AWS DMS da, vedere Migrazione di database compatibili con MySQL a. AWS Se il database di origine non è supportato da AWS DMS, è necessario scegliere un altro metodo per migrare i dati.
- Strumenti MySQL nativi: mysqldump
- Strumenti di terze parti: Percona XtraBackup

Epiche

Pianifica la migrazione

Operazione	Description	Competenze richieste
Convalida le versioni del database.	Convalida le versioni del database di origine e di destinazione.	DBA
Identifica i requisiti hardware.	Identifica i requisiti hardware per il server di destinazione.	DBA, amministratore di sistema
Identifica i requisiti di archiviazione.	Identifica i requisiti di storage (come il tipo e la capacità di storage) per il database di destinazione.	DBA, amministratore di sistema
Scegliere il tipo di istanza.	Scegli il tipo di istanza di destinazione in base alla capacità, alle funzionalità di archiviazione e alle funzionalità di rete.	DBA, amministratore di sistema
Identifica i requisiti di accesso alla rete.	Identifica i requisiti di sicurezza per l'accesso alla rete per i database di origine e di destinazione.	DBA, amministratore di sistema

Operazione	Description	Competenze richieste
Identifica gli oggetti non supportati.	Identifica gli oggetti non supportati (se presenti) e determina lo sforzo di migrazione.	DBA
Identifica le dipendenze.	Identifica eventuali dipendenz e dai database remoti.	DBA
Determinare la strategia di migrazione delle applicazioni.	Determinare la strategia per la migrazione delle applicazioni client.	DBA, proprietario dell'app, amministratore di sistema

Configura l'infrastruttura

Operazione	Description	Competenze richieste
Crea un cloud privato virtuale (Virtual Private Cloud (VPC))	Configura tabelle di routing, gateway Internet, gateway NAT e sottoreti. Per ulteriori informazioni, consulta VPCs Amazon RDS nella documenta zione di Amazon RDS.	Amministratore di sistema
Crea gruppi di sicurezza.	Configura porte e intervall i CIDR o specifici IPs a seconda delle tue esigenze. La porta predefinita per MySQL è 3306. Per ulteriori informazioni, consulta Controllare l'accesso con gruppi di sicurezza nella documentazione di Amazon RDS.	Amministratore di sistema

Operazione	Description	Competenze richieste
Configura e avvia un'istanza DB Amazon RDS for MySQL.	Per istruzioni, consulta Creazione di un'istanza database Amazon RDS nella documentazione di Amazon RDS. Verifica le versioni supportate.	Amministratore di sistema

Migrazione dei dati – opzione 1 (utilizzando strumenti nativi)

Operazione	Description	Competenze richieste
Utilizza strumenti MySQL nativi o strumenti di terze parti per migrare oggetti e dati del database.	Per istruzioni, consulta la documentazione degli strumenti MySQL come mysqldump e Percona (per la migrazione fisica). XtraBackup Per ulteriori informazioni sulle opzioni, consulta il post del blog Opzioni di migrazione per MySQL ad Amazon RDS for MySQL o Amazon Aurora MySQL.	DBA

Migrazione dei dati – opzione 2 (utilizzando) AWS DMS

Operazione	Description	Competenze richieste
Esegui la migrazione dei dati con AWS DMS.	Per istruzioni, consulta la AWS DMS documentazione.	DBA

Esegui le attività preliminari prima del cutover

Operazione	Description	Competenze richieste
Corregge le discrepanze nel conteggio degli oggetti.	Raccogli il conteggio degli oggetti dal database di origine e dal nuovo database di destinazione. Correggi le discrepanze nel database di destinazione.	DBA
Controlla le dipendenze.	Verifica se le dipendenze (collegamenti) da e verso altri database sono valide e funzionano come previsto.	DBA
Esegui dei test.	Se si tratta di un ciclo di test, esegui test delle query, raccogli metriche e risolvi i problemi.	DBA

Tagliare

Operazione	Description	Competenze richieste
Passa al database di destinazione.	Passa le applicazioni client alla nuova infrastruttura.	DBA, proprietario dell'app, amministratore di sistema
Fornisci supporto per i test.	Fornire supporto per i test funzionali delle applicazioni.	DBA

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse.	Chiudi le AWS risorse temporanee che hai creato per la migrazione.	DBA, amministratore di sistema
Convalida i documenti del progetto.	Rivedi e convalida i documenti del progetto.	DBA, proprietario dell'app, amministratore di sistema
Raccogli le metriche.	Raccogli parametri come il tempo necessario per la migrazione, la percentuale di sforzi manuali rispetto a quelli automatizzati, i risparmi sui costi e così via.	DBA, proprietario dell'app, amministratore di sistema
Chiudi il progetto.	Chiudi il progetto e fornisci feedback.	DBA, proprietario dell'app, amministratore di sistema
Disattiva il database di origine.	Una volta completate tutte le attività di migrazione e cutover, disattivate il database locale.	DBA, amministratore di sistema

Risorse correlate

Riferimenti

- Strategia di migrazione per database relazionali
- AWS DMS website
- AWS DMS documentazione
- Documentazione Amazon RDS
- Prezzi di Amazon RDS
- Amazon VPC e Amazon RDS
- Implementazioni Amazon RDS Multi-AZ

- Esegui la migrazione dei database MySQL locali su Aurora MySQL utilizzando Percona, Amazon EFS e Amazon S3 XtraBackup
- Archiviazione delle istanze database Amazon RDS

Tutorial

- Guida introduttiva con AWS DMS
- Nozioni di base su Amazon RDS

Esegui la migrazione di un database Microsoft SQL Server locale su Amazon RDS for SQL Server

Henrique Lobao, Jonathan Pereira Cruz e Vishal Singh, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni per la migrazione da un database Microsoft SQL Server locale ad Amazon Relational Database Service (Amazon RDS) per SQL Server. Descrive due opzioni per la migrazione: utilizzando AWS Data Migration Service (AWS DMS) o utilizzando strumenti nativi di Microsoft SQL Server come Copy Database Wizard.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database Microsoft SQL Server di origine in un data center locale

Limitazioni

- Limite di dimensione del database: 16 TB

Versioni del prodotto

- Per l'elenco più recente delle versioni e delle funzionalità supportate, consulta Microsoft SQL Server su Amazon RDS nella documentazione AWS. Se utilizzi AWS DMS, consulta anche Utilizzo di un database Microsoft SQL Server come destinazione per le versioni di AWS DMS per SQL Server supportate da AWS DMS.

Architecture

Stack tecnologico di origine

- Un database Microsoft SQL Server locale

Stack tecnologico Target

- Un'istanza DB di Amazon RDS per SQL Server

Architettura di origine e destinazione

Utilizzo di AWS DMS:

Utilizzo di strumenti nativi di SQL Server:

Tools (Strumenti)

- AWS DMS supporta diversi tipi di database di origine e destinazione. Per i dettagli, consulta le procedure Step-by-Stepdettagliate di AWS DMS. Se AWS DMS non supporta il database di origine, seleziona un altro metodo per la migrazione dei dati.
- Gli strumenti nativi di Microsoft SQL Server includono il backup e il ripristino, la procedura guidata di copia del database, la copia e il collegamento del database.

Epiche

Pianifica la migrazione

Operazione	Description	Competenze richieste
Convalida la versione e il motore del database di origine e di destinazione.		DBA
Identifica i requisiti hardware per l'istanza del server di destinazione.		DBA, amministratore di sistema
Identifica i requisiti di archiviazione (tipo e capacità di archiviazione).		DBA, amministratore di sistema
Scegli il tipo di istanza corretto in base alla capacità, alle funzionalità di archiviazione e alle funzionalità di rete.		DBA, amministratore di sistema

Operazione	Description	Competenze richieste
Identifica i requisiti di sicurezza dell'accesso alla rete per i database di origine e di destinazione.		DBA, amministratore di sistema
Identifica la strategia di migrazione delle applicazioni.		DBA, amministratore di sistema

Configura l'infrastruttura

Operazione	Description	Competenze richieste
Crea un cloud privato virtuale (Virtual Private Cloud (VPC))		Amministratore di sistema
Crea gruppi di sicurezza.		Amministratore di sistema
Configura e avvia un'istanza database Amazon RDS.		DBA, amministratore di sistema

Migrazione dei dati - opzione 1

Operazione	Description	Competenze richieste
Utilizza strumenti nativi di SQL Server o strumenti di terze parti per migrare oggetti e dati del database.		DBA

Migrazione dei dati - opzione 2

Operazione	Description	Competenze richieste
Migra i dati con AWS DMS.		DBA

Migrare l'applicazione

Operazione	Description	Competenze richieste
Segui la strategia di migrazione delle applicazioni.		DBA, proprietario dell'app, amministratore di sistema

Tagliare

Operazione	Description	Competenze richieste
Trasferisci i client applicativi alla nuova infrastruttura.		DBA, proprietario dell'app, amministratore di sistema

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse AWS temporanee.		DBA, amministratore di sistema
Rivedi e convalida i documenti del progetto.		DBA, proprietario dell'app, amministratore di sistema
Raccogli parametri come il tempo necessario per la migrazione, la percentuale di attività manuali rispetto		DBA, proprietario dell'app, amministratore di sistema

Operazione	Description	Competenze richieste
a quelle automatizzate e il risparmio sui costi.		
Chiudi il progetto e fornisci feedback.		DBA, proprietario dell'app, amministratore di sistema

Risorse correlate

Riferimenti

- Implementazione di Microsoft SQL Server su Amazon Web Services
- Sito web AWS DMS
- Prezzi di Amazon RDS
- Prodotti Microsoft su AWS
- Licenze Microsoft su AWS
- Microsoft SQL Server su AWS
- Utilizzo dell'autenticazione di Windows con un'istanza DB di Microsoft SQL Server
- Implementazioni Multi-AZ di Amazon RDS

Tutorial e video

- Guida introduttiva ad AWS DMS
- Nozioni di base su Amazon RDS
- AWS DMS (video)
- Amazon RDS (video)

Esegui la migrazione dei dati da Microsoft Azure Blob ad Amazon S3 utilizzando Rclone

Suhas Basavaraj, Aidan Keane e Corey Lane, Amazon Web Services

Riepilogo

Questo modello descrive come usare Rclone per migrare i dati dallo storage di oggetti Microsoft Azure Blob a un bucket Amazon Simple Storage Service (Amazon S3). È possibile utilizzare questo modello per eseguire una migrazione una tantum o una sincronizzazione continua dei dati. Rclone è un programma a riga di comando scritto in Go e viene utilizzato per spostare i dati tra varie tecnologie di archiviazione dei provider di cloud.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Dati archiviati nel servizio contenitore Azure Blob

Architecture

Stack tecnologico di origine

- Contenitore di archiviazione Azure Blob

Stack tecnologico Target

- Bucket Amazon S3
- Istanza Amazon Elastic Compute Cloud (Amazon EC2) Linux

Architettura

Tools (Strumenti)

- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

- Rclone è un programma a riga di comando open source ispirato a rsync. Viene utilizzato per gestire i file su molte piattaforme di archiviazione cloud.

Best practice

Quando esegui la migrazione dei dati da Azure ad Amazon S3, tieni presente queste considerazioni per evitare costi inutili o velocità di trasferimento lente:

- Crea la tua infrastruttura AWS nella stessa regione geografica dell'account di archiviazione di Azure e del contenitore Blob, ad esempio, la `us-east-1` regione AWS (Virginia settentrionale) e la regione Azure. `East US`
- Se possibile, evita di usare NAT Gateway, perché comporta costi di trasferimento dei dati sia per la larghezza di banda in ingresso che in uscita.
- Usa un endpoint gateway VPC per Amazon S3 per aumentare le prestazioni.
- Prendi in considerazione l'utilizzo di un' EC2 istanza basata su processore AWS Graviton2 (ARM) per costi inferiori e prestazioni più elevate rispetto alle istanze Intel x86. Rclone è fortemente compilato in modo incrociato e fornisce un binario ARM precompilato.

Epiche

Prepara le risorse cloud AWS e Azure

Operazione	Description	Competenze richieste
Prepara un bucket S3 di destinazione.	Crea un nuovo bucket S3 nella regione AWS appropriata o scegli un bucket esistente come destinazione per i dati che desideri migrare.	Amministratore AWS
Crea un ruolo di istanza IAM per Amazon EC2.	Crea un nuovo ruolo AWS Identity and Access Management (IAM) per Amazon EC2. Questo ruolo fornisce all' EC2 istanza l'accesso in scrittura al bucket S3 di destinazione.	Amministratore AWS

Operazione	Description	Competenze richieste
Allega una policy al ruolo dell'istanza IAM.	Utilizza la console IAM o AWS Command Line Interface (AWS CLI) per creare una policy in linea per EC2 il ruolo dell'istanza che consente le autorizzazioni di accesso in scrittura al bucket S3 di destinazione. Per un esempio di policy, consulta la sezione Informazioni aggiuntive.	Amministratore AWS
Avvia un' EC2 istanza.	Avvia un' EC2 istanza Amazon Linux configurata per utilizzare il ruolo di servizio IAM appena creato. Questa istanza dovrà inoltre accedere agli endpoint delle API pubbliche di Azure tramite Internet. **Note** Prendi in considera zione l'utilizzo di EC2 istanze basate su AWS Graviton per ridurre i costi. Rclone fornisce file binari compilati da ARM.	Amministratore AWS

Operazione	Description	Competenze richieste
Crea un responsabile del servizio Azure AD.	Usa l'interfaccia della riga di comando di Azure per creare un servizio principal e di Azure Active Directory (Azure AD) con accesso in sola lettura al contenitore di archiviazione Azure Blob di origine. Per istruzioni, vedere la sezione Informazioni aggiuntive. Archivia queste credenziali sulla tua EC2 istanza nella posizione `~/azure-principal.json` .	Amministratore cloud, Azure

Installa e configura Rclone

Operazione	Description	Competenze richieste
Scarica e installa Rclone.	Scarica e installa il programma da riga di comando Rclone. Per le istruzioni di installazione, consulta la documentazione di installazione di Rclone.	AWS generale, amministratore del cloud
Configura Rclone.	Copia il seguente file di `rclone.conf` esempio. `AZStorage Account` Sostituiscilo con il nome del tuo account di archiviazione di Azure e `us-east-1` con la regione AWS in cui si trova il bucket S3. Salva questo	AWS generale, amministratore del cloud

Operazione	Description	Competenze richieste

	file nella posizione dell'istanza `~/.config/rclone/rclone.conf` . EC2
	
	`[AZStorageAccount]`
	`type = azureblob`
	`account = AZStorageAccount`
	`service_principal_file = azure-principal.json`
	
	`[s3]`
	`type = s3`
	`provider = AWS`
	`env_auth = true`
	`region = us-east-1`	

Operazione	Description	Competenze richieste

Verifica la configurazione di Rclone.	Per confermare che Rclone sia configurato e che le autorizzazioni funzionino correttamente, verifica che Rclone sia in grado di analizzare il file di configurazione e che gli oggetti all'interno del contenitore Azure Blob e del bucket S3 siano accessibili. Vedi quanto segue, ad esempio, i comandi di convalida.
	• Elenca i telecomandi configurati nel file di configurazione. Ciò garantirà che il file di configurazione venga analizzato correttamente. Controlla l'output per assicurarti che corrisponda al tuo `rclone.conf` file.
	`rclone listremotes`
	`AZStorageAccount:`
	`s3:`
	• Elenca i contenitori Azure Blob nell'account configurato. Sostituiscili `AZStorageAccount` con il nome dell'account di archiviazione che hai usato nel `rclone.conf` file.
		AWS generale, amministratore del cloud

Operazione	Description	Competenze richieste

	`rclone lsd AZStorage Account:`
	`2020-04-29 08:29:26 docs`
	• Elenca i file nel contenitore Azure Blob. Sostituisci i documenti in questo comando con un nome effettivo del contenitore Blob nel tuo account di archiviazione di Azure.
	`rclone ls AZStorage Account:docs`
	`824884 administrator-en.a4.pdf`
	• Elenca i bucket nel tuo account AWS.
	`[root@ip-10-0-20-157 ~]# rclone lsd s3:`
	`2022-03-07 01:44:40 amzn-s3-demo-bucket1`
	`2022-03-07 01:45:16 amzn-s3-demo-bucket2`
	`2022-03-07 02:12:07 amzn-s3-demo-bucket3`
	• Elenca i file nel bucket S3.
	`[root@ip-10-0-20-157 ~]# rclone ls s3:amzn-s3-demo-bucket1`	

Operazione	Description	Competenze richieste
	`template0.yaml` `template1.yaml`	

Migra i dati utilizzando Rclone

Operazione	Description	Competenze richieste

Migra i dati dai tuoi contenitori.	Esegui il comando Rclone copy or sync.
	
	Esempio: copia
	
	Questo comando copia i dati dal contenitore Azure Blob di origine al bucket S3 di destinazione.
	
	`rclone copy AZStorageAccount:blob-container s3:amzn-s3-demo-bucket1`
	
	Esempio: sync
	
	Questo comando sincronizza i dati tra il contenitore Azure Blob di origine e il bucket S3 di destinazione.
	
	`rclone sync AZStorageAccount:blob-container s3:amzn-s3-demo-bucket1`
		AWS generale, amministratore del cloud

Operazione	Description	Competenze richieste
	Important Quando usi il comando sync, i dati che non sono presenti nel contenitore di origine verranno eliminati dal bucket S3 di destinazione.	
Sincronizza i tuoi contenitori.	Una volta completata la copia iniziale, esegui il comando Rclone sync per la migrazione in corso in modo che vengano copiati solo i nuovi file mancanti dal bucket S3 di destinazione.	AWS generale, amministratore del cloud
Verifica che i dati siano stati migrati correttamente.	Per verificare che i dati siano stati copiati correttamente nel bucket S3 di destinazione, esegui i comandi Rclone lsd e ls.	AWS generale, amministratore del cloud

Risorse correlate

- Guida per l'utente di Amazon S3 (documentazione AWS)
- Ruoli IAM per Amazon EC2 (documentazione AWS)
- Creazione di un contenitore Microsoft Azure Blob (documentazione di Microsoft Azure)
- Comandi Rclone (documentazione Rclone)

Informazioni aggiuntive

Esempio di politica di ruolo per le istanze EC2

Questa politica offre all' EC2 istanza l'accesso in lettura e scrittura a un bucket specifico del tuo account. Se il bucket utilizza una chiave gestita dal cliente per la crittografia lato server, la policy potrebbe richiedere un accesso aggiuntivo ad AWS Key Management Service (AWS KMS).

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket",
                "s3:DeleteObject",
                "s3:GetObject",
                "s3:PutObject",
                "s3:PutObjectAcl"
            ],
            "Resource": [
              "arn:aws:s3:::amzn-s3-demo-bucket/*",
              "arn:aws:s3:::amzn-s3-demo-bucket"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "s3:ListAllMyBuckets",
            "Resource": "arn:aws:s3:::*"
        }
    ]
}
```

Creazione di un principale di servizio Azure AD di sola lettura

Un service principal di Azure è un'identità di sicurezza usata dalle applicazioni, dai servizi e dagli strumenti di automazione dei clienti per accedere a risorse di Azure specifiche. Pensala come un'identità utente (login e password o certificato) con un ruolo specifico e autorizzazioni strettamente controllate per accedere alle tue risorse. Per creare un servizio principale di sola lettura che utilizzi le autorizzazioni con privilegi minimi e protegga i dati in Azure da eliminazioni accidentali, segui questi passaggi:

1. Accedi al portale del tuo account cloud Microsoft Azure e avvia Cloud Shell PowerShell o usa l'interfaccia a riga di comando (CLI) di Azure sulla tua workstation.

2. Crea un service principal e configuralo con accesso in sola lettura al tuo account di archiviazione Azure Blob. Salva l'output JSON di questo comando in un file locale chiamato. `azure-principal.json` Il file verrà caricato sulla tua EC2 istanza. Sostituisci le variabili segnaposto mostrate tra parentesi quadre (`{e}`) con l'ID di sottoscrizione di Azure, il nome del gruppo di risorse e il nome dell'account di archiviazione.

```
az ad sp create-for-rbac `
--name AWS-Rclone-Reader `
--role "Storage Blob Data Reader" `
--scopes /subscriptions/{Subscription ID}/resourceGroups/{Resource Group Name}/
providers/Microsoft.Storage/storageAccounts/{Storage Account Name}
```

Migrazione da Couchbase Server a Couchbase Capella su AWS

Battulga Purevragcha e Saurabh Shanbhag, Amazon Web Services

Mark Gamble, Nessuno

Riepilogo

Couchbase Capella è un database NoSQL come servizio (DBaaS) completamente gestito per applicazioni mission critical (ad esempio, profili utente o cataloghi online e gestione dell'inventario). Couchbase Capella gestisce il tuo carico di lavoro DBaa S in un account Amazon Web Services (AWS) gestito da Couchbase. Capella semplifica l'esecuzione e la gestione della replica su più cluster, più regioni AWS, multicloud e cloud ibrido all'interno di un'unica interfaccia.

Couchbase Capella ti aiuta a scalare istantaneamente le tue applicazioni Couchbase Server, aiutandoti a creare cluster multinodo in pochi minuti. Couchbase Capella supporta tutte le funzionalità di Couchbase Server, tra cui SQL++, Full Text Search, Eventing Service e Analytics Service. Inoltre elimina la necessità di gestire installazioni, aggiornamenti, backup e manutenzione generale del database.

Questo modello descrive i passaggi e le migliori pratiche per la migrazione di un ambiente Couchbase Server autogestito al cloud AWS. Il modello fornisce un processo ripetibile per la migrazione di dati e indici dai cluster di Couchbase Server, in esecuzione in locale o nel cloud, a Couchbase Capella. L'utilizzo di questi passaggi consente di evitare problemi durante la migrazione e velocizza il processo di migrazione complessivo.

Questo modello fornisce le due opzioni di migrazione seguenti:

- L'opzione 1 è appropriata se avete meno di 50 indici da migrare.
- L'opzione 2 è appropriata se avete più di 50 indici da migrare.

Puoi anche configurare dati di esempio sul tuo Couchbase Server autogestito da seguire insieme alla guida alla migrazione.

Se scegli l'opzione di migrazione 2 o se utilizzi ambiti o raccolte diversi dal valore predefinito, devi utilizzare il file di configurazione di esempio, che si trova nella sezione Informazioni aggiuntive.

Prerequisiti e limitazioni

Prerequisiti

- Un account a pagamento Couchbase Capella esistente. Puoi anche creare un account Couchbase Capella su AWS e utilizzare la versione di prova gratuita di Couchbase Capella, quindi passare a un account a pagamento per configurare il cluster per la migrazione. Per iniziare con la versione di prova, segui le istruzioni contenute in Getting Started with Couchbase Capella.
- Un ambiente Couchbase Server esistente autogestito in locale o distribuito su un provider di servizi cloud.
- Per l'opzione di migrazione 2, Couchbase Shell e un file di configurazione. Per creare il file di configurazione, puoi utilizzare il file di esempio che si trova nella sezione Informazioni aggiuntive.
- Familiarità con l'amministrazione di Couchbase Server e Couchbase Capella.
- Familiarità con l'apertura di porte TCP e l'esecuzione di comandi in un'interfaccia a riga di comando (CLI).

Il processo di migrazione richiede anche i ruoli e le competenze descritti nella tabella seguente.

Ruolo	Competenza	Responsabilità
Amministratore di Couchbase	• Familiarità con Couchbase Server e Couchbase Capella • La conoscenza di base della riga di comando è utile ma non richiesta	• Attività specifiche di Couchbase Server e Capella
Amministratore di sistema, amministratore IT	• Familiarità con l'ambient e e l'amministrazione del sistema Couchbase Server autogestiti	• Apertura delle porte e determinazione degli indirizzi IP sui nodi del cluster Couchbase Server autogestiti

Limitazioni

- Questo modello viene utilizzato per migrare dati, indici e indici Couchbase Full Text Search da Couchbase Server a Couchbase Capella su AWS. Il modello non si applica alla migrazione di Couchbase Eventing Service o a Couchbase Analytics.

- Couchbase Capella è disponibile in diverse regioni AWS. Per up-to-date informazioni sulle regioni supportate da Capella, consulta Amazon Web Services nella documentazione di Couchbase.

Versioni del prodotto

- Couchbase Server (Community o Enterprise) Edition versione 5.x o successiva

Architecture

Stack tecnologico di origine

- Server Couchbase

Stack tecnologico Target

- Divano Capella

Architettura Target

1. Puoi accedere a Couchbase Capella utilizzando il Capella Control Plane. È possibile utilizzare il Capella Control Plane per effettuare le seguenti operazioni:
   - Controlla e monitora il tuo account.
   - Gestisci cluster e dati, indici, utenti e gruppi, autorizzazioni di accesso, monitoraggio ed eventi.
2. I cluster vengono creati.
3. Il Capella Data Plane si trova nell'account AWS gestito da Couchbase. Dopo aver creato un nuovo cluster, Couchbase Capella lo distribuisce su più zone di disponibilità nella regione AWS selezionata.
4. Puoi sviluppare e distribuire applicazioni Couchbase in un VPC nel tuo account AWS. In genere, questo VPC accede al Capella Data Plane tramite peering VPC.

Tools (Strumenti)

- Couchbase Cross Data Center Replication (XDCR) aiuta a replicare i dati tra cluster che si trovano in diversi provider di cloud e diversi data center. Viene utilizzato per migrare i dati in Couchbase Capella da cluster Couchbase Server autogestiti.

> **Note**
>
> XDCR non può essere utilizzato con Couchbase Server Community Edition per migrare a Couchbase Capella. Invece, puoi usare cbexport. Per ulteriori informazioni, consulta l'epic Migrazione dei dati dalla Community Edition.

- Couchbase Shell è una shell a riga di comando per Couchbase Server e Couchbase Capella per accedere ai cluster Couchbase locali e remoti. In questo modello, Couchbase Shell viene utilizzata per migrare gli indici.
- cbexport è un'utilità Couchbase per l'esportazione di dati dal cluster Couchbase. Incluso negli strumenti CLI di Couchbase Server.

Epiche

Prepara la migrazione

Operazione	Description	Competenze richieste
Valuta le dimensioni del cluster Couchbase Server autogestito.	Accedi alla Couchbase Web Console per Couchbase Server e valuta i nodi e i bucket del cluster autogestito. 1. Per mostrare un elenco di nodi del cluster, scegli la scheda Server nella barra di navigazione. 2. Registra il numero di nodi, quindi scegli ogni nodo nell'elenco per visualizzarne le proprietà. 3. Registra la memoria e lo storage per ogni singolo nodo. 4. Scegli la scheda Bucket nella barra di navigazione,	Amministratore di Couchbase

Operazione	Description	Competenze richieste
	quindi scegli ogni bucket nell'elenco per visualizz arne le proprietà. Registra la quota di RAM e l'imposta zione di risoluzione dei conflitti per ogni bucket. Utilizzerai le configurazioni del cluster Couchbase Server autogestite come guida generale per il dimension amento e la configurazione del cluster di destinazione su Couchbase Capella. Per assistenza con un esercizio di dimension amento di Couchbase Capella più dettagliato, contatta Couchbase.	

Operazione	Description	Competenze richieste
Registra la distribuzione del servizio Couchbase sul cluster Couchbase Server autogesti to.	1. Nella Couchbase Web Console, scegli la scheda Server per visualizzare l'elenco dei nodi del cluster. 2. Scegli ogni nodo per visualizzarne le proprietà e quindi registra la distribuzione del servizio Couchbase per ogni nodo (Data Service, Query Service, Index Service, Search Service, Analytics Serviceed Eventing Service).	Amministratore di Couchbase
Registra gli indirizzi IP dei nodi del cluster Couchbase Server autogestiti.	(Ignora questo passaggio se utilizzi Community Edition.) Registra l'indirizzo IP per ogni nodo del cluster. Verranno aggiunti all'elenco degli utenti consentiti sul cluster Couchbase Capella in un secondo momento.	Amministratore di Couchbase, amministratore di sistema

Distribuisci e configura le risorse su Couchbase Capella

Operazione	Description	Competenze richieste
Scegliere un modello.	1. Accedi al tuo Couchbase Capella Control Plane, scegli la scheda Dashboard o la scheda Clusters nella	Amministratore di Couchbase

Operazione	Description	Competenze richieste
	navigazione principale, quindi scegli Crea cluster. 2. Utilizzando le informazioni che hai registrato dalla valutazione del cluster Couchbase Server autogestito, scegli il modello di cluster che soddisfa i requisiti della configurazione. Se non trovi un modello appropriato, scegli Modello personalizzato nell'editor di dimensionamento del cluster.	
Scegli e configura i nodi.	Scegli e configura i nodi in modo che corrispondano al tuo ambiente cluster Couchbase Server autogestito, inclusi il numero di nodi, la distribuzione dei servizi, l'elaborazione o la RAM e lo storage. Couchbase Capella utilizza le migliori pratiche di scalabilità multidimensionale. I servizi e i nodi possono essere scelti solo in base alle migliori pratiche di implementazione. Ciò potrebbe significare che non è possibile abbinare esattamente le configurazioni del cluster Couchbase Server autogestito.	Amministratore di Couchbase

Operazione	Description	Competenze richieste
Implementa il cluster.	Scegli una zona di supporto e un pacchetto di supporto, quindi distribuisci il cluster. Per passaggi e istruzioni dettagliati, consulta Creare un cluster nella documentazione di Couchbase. **Important** Se utilizzi la versione di prova gratuita di Couchbase Capella, devi convertirla in un account a pagamento prima di iniziare la migrazione. Per convertire il tuo account, apri la sezione Fatturazione del piano di controllo di Couchbase Capella, quindi scegli Aggiungi ID di attivazione. L'ID di attivazione viene inviato al tuo indirizzo email di contatto per la fatturazione dopo aver completato un contratto di acquisto con Couchbase Sales o dopo aver effettuato un acquisto tramite AWS Marketplace.	Amministratore di Couchbase

Operazione	Description	Competenze richieste
Crea un utente di credenziali del database.	Un utente di credenziali del database è specifico di un cluster ed è composto da un nome utente, una password e un set di privilegi del bucket. Questo utente è necessario per creare bucket e accedere ai dati dei bucket. Nel piano di controllo di Couchbase Capella, crea una credenziale di database per il nuovo cluster seguendo le istruzioni in Configurare le credenziali del database nella documentazione di Couchbase Capella. **Note** A un utente dell'organizzazione devono essere assegnate le credenziali del ruolo organizzativo se desidera accedere ai dati dei bucket su un particolare cluster, in remoto o tramite l'interfaccia utente di Couchbase Capella. Questo è separato dalle credenziali del database, che vengono in genere	Amministratore di Couchbase

Operazione	Description	Competenze richieste
	utilizzate da app e integrazioni. La creazione dell'uten te organizzativo ti consente di creare e gestire i bucket di destinazione sul tuo cluster Couchbase Capella.	
Se utilizzi l'opzione di migrazione 2, installa Couchbase Shell.	Puoi installare Couchbase Shell su qualsiasi sistema che abbia accesso di rete sia al tuo Couchbase Server autogestito che ai cluster Couchbase Capella. Per ulteriori informazioni, consulta Installa Couchbase Shell versione 1.0.0-beta.5 nella documentazione di Couchbase Shell. Verifica che Couchbase Shell sia installato testando una connessione al cluster autogestito in un terminale a riga di comando.	Amministratore di Couchbase, amministratore di sistema

Operazione	Description	Competenze richieste
Consenti indirizzi IP.	1. Nel piano di controllo di Couchbase Capella, scegli Clusters, quindi scegli il cluster di destinazione. 2. Scegli la scheda Connect per il cluster e registra l'endpoint di connessio ne per il tuo cluster che è elencato in Gestisci IP consentito. 3. Per aggiungere l'indirizzo IP del sistema su cui hai installato Couchbase Shell e l'indirizzo IP delle istanze del cluster Couchbase Server autogestite come indirizzi IP consentiti, procedi come segue: a. In Wide Area Network, scegli Gestisci IP consentito. b. Scegli Aggiungi IP consentito, inserisci l'indirizzo IP del sistema su cui hai installato Couchbase Shell, quindi scegli Aggiungi IP. c. Ripeti il passaggio precedente per aggiungere l'indirizzo IP dell'istanza del cluster Couchbase Server autogestita.	Amministratore di Couchbase, Amministratore di sistema

Operazione	Description	Competenze richieste
	Per ulteriori informazioni sugli indirizzi IP consentiti, consulta Configurare gli indirizzi IP consentiti nella documenta zione di Couchbase.	
Configura i certificati.	1. Per scaricare il certificato radice per il cluster, in Root Certificate, scegli Scarica. 2. Salva il certificato principal e utilizzando l'estensione del file.pem in una cartella del sistema che eseguirà Couchbase Shell. 3. Successivamente, accedi alla console Web di Couchbase Server autogestita, scegli Sicurezza nella barra di navigazione a sinistra, quindi scegli la scheda Certificati. 4. Copia il certificato principal e per il cluster Couchbase Server autogestito e salvalo come file.pem nella stessa cartella in cui hai salvato il file del certificato radice per il tuo cluster Couchbase Capella. Per ulteriori informazioni sul certificato root, consulta Root certifica te nella documentazione di Couchbase Server.	Amministratore di Couchbase, amministratore di sistema

Operazione	Description	Competenze richieste
Crea il file di configurazione per Couchbase Shell.	Crea un dotfile di configura zione nella home directory dell'installazione di Couchbase Shell (ad esempio,). `/<HOME_DIRECTORY>/.cbsh/config` Per ulteriori informazioni, consulta Config dotfiles nella documentazione di Couchbase. Aggiungi le proprietà di connessione per i cluster di origine e di destinazione al file di configurazione. Puoi utilizzare il file di configura zione di esempio che si trova nella sezione Informazioni aggiuntive e modificare le impostazioni per i tuoi cluster. Salva il file di configurazione con le impostazioni aggiornat e nella `.cbsh` cartella (ad esempio,`/<HOME_DIRECTORY>/.cbsh/config`).	Amministratore di Couchbase, amministratore di sistema

Operazione	Description	Competenze richieste
Crea bucket di destinazione.	Per ogni bucket di origine, crea un bucket di destinazione nel cluster Couchbase Capella seguendo le istruzioni in Creare un bucket nella documentazione di Couchbase. Le configurazioni dei bucket di destinazione devono corrispondere ai nomi dei bucket, alle impostazioni di memoria e alle impostazioni di risoluzione dei conflitti dei bucket nel cluster Couchbase Server autogestito.	Amministratore di Couchbase

Operazione	Description	Competenze richieste

Crea ambiti e raccolte.	Ogni bucket contiene un ambito e una raccolta predefiniti con lo spazio delle chiavi. `_default._default` Se utilizzi altri spazi chiave per l'ambito e la raccolta, devi creare spazi chiave identici nel cluster Capella di destinazione.
	1. Apri il terminale a riga di comando sul sistema in cui hai installato Couchbase Shell.
	2. Per avviare Couchbase Shell, esegui il seguente comando.
	`./cbsh`
	3. Per ogni bucket che desideri migrare, crea ambiti e raccolte nel cluster Capella eseguendo i seguenti comandi. Assicurati di sostituirlo <BUCKET_NAME>con il nome del bucket che desideri migrare.
	`scopes --clusters "On-Prem-Cluster" --bucket <BUCKET_NAME> | select scope | where scope != "_default" | each`
		Amministratore di Couchbase

Operazione	Description	Competenze richieste
	`{ |it| scopes create $it.scope --clusters "Capella-Cluster" } collections --clusters "On-Prem-Cluster" --bucket <BUCKET_NAME> | select scope collection | where $it.scope != "_default" | where $it.collection != "_default" | each { |it| collections create $it.collection --clusters "Capella-Cluster" --bucket <BUCKET_NAME> --scope $it.scope }`	

Esegui la migrazione dei dati da Enterprise Edition

Operazione	Description	Competenze richieste
Apri le porte TCP sui nodi del cluster Couchbase Server autogestiti.	Assicurati che le porte appropriate siano aperte per la comunicazione XDCR sui nodi del cluster Couchbase Server autogestito. Per ulteriori informazioni, consulta la documentazione sulle porte del server Couchbase.	Amministratore di Couchbase, amministratore di sistema
Se utilizzi Couchbase Server Enterprise Edition, configura Couchbase XDCR.	1. Nella navigazione principale di Couchbase Capella Control Plane, scegli Clusters, quindi scegli il cluster di destinazione per la migrazione.	Amministratore di Couchbase

Operazione	Description	Competenze richieste
	2. In Root Certificate, scegli Copia. 3. Accedi alla console Web di Couchbase Server autogestita e, nella navigazione principale, scegli XDCR. Quindi scegli Aggiungi remoto. 4. Specificare le seguenti impostazioni: • Nome cluster: un nome per la connessione al cluster Capella • IP/hostname: l'endpoint di connessione per il cluster Couchbase Capella • Nome utente per Remote Cluster: l'utente del database per il cluster Couchbase Capella • Password: la password utente del database per il cluster Couchbase Capella • Abilita connessione sicura: selezionato • Completo (password e dati crittografati con TLS): selezionato 5. Incolla il certificato radice del cluster Capella che hai copiato in precedenza, quindi scegli Salva.	

Operazione	Description	Competenze richieste
Avvia Couchbase XDCR.	1. Nella console Web di Couchbase Server autogestita, scegli XDCR nella navigazione principal e, quindi scegli Aggiungi replica. 2. Specificare le seguenti impostazioni: • Replicate From Bucket: seleziona il bucket di origine per la migrazione. • Bucket remoto: immetti il nome del bucket di destinazione. • Cluster remoto: seleziona il cluster di destinazione creato in precedenza. 3. Scegli Salva replica. Il processo di replica dovrebbe iniziare entro pochi secondi.	Amministratore di Couchbase

Migra gli indici utilizzando l'opzione 1

Operazione	Description	Competenze richieste
Migra gli indici di cluster autogestiti su Couchbase Capella.	⚠ Important Consigliamo questa procedura se hai meno di 50 indici da migrare. Se hai più di 50 indici da	Amministratore di Couchbase, amministratore di sistema

Operazione	Description	Competenze richieste
	migrare, ti consigliamo di utilizzare l'opzione di migrazione 2. 1. Nella console Web Couchbase, scegli Indici. 2. Nell'elenco degli indici, scegli il primo indice che desideri migrare. Viene quindi visualizzata la definizione dell'indice. 3. Copiate la definizione dell'indice utilizzando l'`CREATE`istruzione, ma non copiatela `WITH { "defer_build":true }` . Ad esempio, dall'esempio seguente è possibile copiare solo la definizione dell'indice`CREATE INDEX` `` `cityindex` `` `ON` `` `travel-sample`(`city`) `` . ``` CREATE INDEX `cityindex` ON `travel-sample`(`city`) WITH { "defer_build":true } ``` 4. Nel piano di controllo Couchbase Capella, scegli	

Operazione	Description	Competenze richieste
	Clusters, quindi scegli il cluster di destinazione. 5. Nell'elenco a discesa Strumenti, scegli Query Workbench. Incolla l'**`CREATE`**istruzione che hai copiato in precedenz a nel Query Editor, quindi scegli Esegui. Questo crea e costruisce l'indice. 6. Per confermare la creazione dell'indice, scegli Indici dall'elenco a discesa Strumenti. L'elenco mostra che l'indice è stato creato e creato. 7. Ripeti questo processo per ogni indice che deve essere migrato.	

Esegui la migrazione degli indici utilizzando l'opzione 2

Operazione	Description	Competenze richieste
Esegui la migrazione delle definizioni degli indici.	⚠ Important Consigliamo questa procedura se hai più di 50 indici da migrare. Se hai meno di 50 indici da migrare, ti consigliamo di	Amministratore di Couchbase, amministratore di sistema

Operazione	Description	Competenze richieste

	utilizzare l'opzione di migrazione 1.
	
	1. Apri il terminale a riga di comando sul sistema in cui hai installato Couchbase Shell.
	2. Per avviare Couchbase Shell, esegui il seguente comando.
	
	`./cbsh`
	
	3. Per connetterti al cluster Couchbase Server autogestito, esegui il comando seguente.
	
	`cb-env cluster On-Prem-Cluster`
	
	4. Per migrare le definizioni degli indici dal cluster Couchbase Server autogestito al cluster Couchbase Capella, esegui il comando seguente per ogni bucket che desideri migrare. Assicurati di sostituirlo <BUCKET_NAME> con il nome del bucket che corrisponde agli indici che desideri migrare. Questa opzione di migrazione richiede che i	

Operazione	Description	Competenze richieste
	nomi dei bucket di destinazione siano identici ai nomi dei bucket di origine. `query indexes --definitions | where bucket =~ <BUCKET_NAME>| get definition | each { |it| query $it --clusters Capella-Cluster }`	

Operazione	Description	Competenze richieste

Crea le definizioni degli indici.	1. Per passare dal contesto al cluster Couchbase Capella, esegui il seguente comando:
	`cb-env cluster Capella-Cluster`
	2. Per creare le definizioni degli indici che sono state migrate nel cluster Couchbase Capella, esegui il comando seguente, sostituendolo <BUCKET_NAME> con il nome del bucket corrispondente agli indici che desideri creare.
	`query 'SELECT RAW CONCAT("BUILD INDEX ON ", k , "(['", CONCAT2 ("','", inames), "']);") FROM system:indexes AS s LET bid = CONCAT("`",s.bucket_id, "`"), sid = CONCAT("`", s.scope_id, "`"), kid = CONCAT("`", s.keyspace_id, "`"), k = NVL2(bid, CONCAT2(".", bid, sid, kid), kid) WHERE s.namespace_id = "default" AND s.bucket_id = "" GROUP BY k LETTING inames = ARRAY_AGG(s.name) FILTER`
		Amministratore di Couchbase, amministratore di sistema

Operazione	Description	Competenze richieste
	(WHERE s.state = 'deferred') HAVING ARRAY_LENGTH(iname s) > 0;' | each { | it| query $it } 3. Ripetere l'operazione per ogni bucket.	

Migrazione degli indici di ricerca full-text

Operazione	Description	Competenze richieste
Migra gli indici di ricerca full-text del cluster autogestiti su Couchbase Capella.	1. Nella Couchbase Web Console, scegli Cerca. 2. Nell'elenco degli indici di ricerca full-text (FTS), scegli il primo indice FTS che desideri migrare, scegli Mostra la definizione dell'indice JSON e scegli Copia negli Appunti. Prendi nota del nome dell'indice e del bucket a cui appartiene. 3. Nel Couchbase Capella Control Plane, scegli Clusters, quindi scegli il cluster di destinazione. 4. Nell'elenco a discesa Strumenti, scegli Ricerca nel testo completo. 5. Scegli Importa indice e incolla la definizione dell'indice FTS.	Amministratore di Couchbase

Operazione	Description	Competenze richieste
	6. Inserisci il nome dell'indice, seleziona il bucket corretto, come indicato nel cluster autogestito, quindi scegli Crea. 7. Ripeti questo processo per ogni indice FTS che deve essere migrato.	

Migra i dati da Couchbase Community Edition

Operazione	Description	Competenze richieste

Esporta i dati da Couchbase Server Community Edition autogestito.	L'XDCR crittografato non è disponibile in Couchbase Community Edition. È possibile esportare i dati da Couchbase Community Edition e quindi importarli manualmente in Couchbase Capella.
	
	Per esportare i dati dal bucket di origine, usali nella riga di comando. `cbexport`
	
	Il comando seguente viene fornito come esempio.
	
	`cbexport json \`
	`--cluster localhost \`
	`--bucket <SOURCE BUCKET NAME> \`
	`--format lines \`
	`--username <USERNAME> \`
		Amministratore di Couchbase

Operazione	Description	Competenze richieste

	`--password <PASSWORD> \`
	`--include-key cbkey \`
	`--scope-field cbscope \`
	`--collection-field cbcoll \`
	`--output cbexported_data.json`
	
	Nota che `cbkey`, `cbscope` e `cbcoll`, e `cbexported_data.json` sono etichette arbitrarie. Verrà fatto riferimento a esse più avanti nel processo, quindi se scegli di denominarle in modo diverso, prendine nota.	

Operazione	Description	Competenze richieste
Importa i dati in Couchbase Capella.	1. Nel piano di controllo di Couchbase Capella, scegli Clusters, quindi scegli il cluster di destinazione. 2. Nell'elenco a discesa Strumenti, scegli Importa. Si aprirà una procedura guidata con i seguenti sei passaggi: a. Bucket: scegli il bucket di destinazione. b. File: scegli JSON, scegli Lines, quindi scegli Utilizzo del browser web. Se disponi di una grande quantità di dati, puoi esplorare l'opzione Manualmente. Seleziona il file creato `dacbexport`. c. Raccolte: scegli la mappatura personali zzata delle raccolte. Se il tuo database Community Edition non utilizza ambiti o raccolte o utilizza solo _default, puoi scegliere invece l'opzione Seleziona raccolta singola. Per Collection Mapping Expression, immettere.	Amministratore di Couchbase

Operazione	Description	Competenze richieste
	`%cbscope%.%cbcoll%` Per verificare che questa espressione funzioni correttamente, è possibile incollare dati di esempio, come i seguenti. `{ "cbscope" :"inventory", "cbcoll":"landmark", "cbkey":" landmark_3991" }` d. Chiave: scegli Customer Generation. (Se non ti interessa conservare le chiavi dei dati che stai importando, puoi invece selezionare UUID generato automaticamente e procedere al passaggio 5.) Per Key Name Generator Expression, inserisci %cbkey%. Per verificare che questa espressione funzioni correttamente, incolla alcuni dati di esempio. e. Configurazioni: scegli Ignora campi e inserisci cbscope, cbcoll, cbkey. Questi campi contengono informazioni transitorie che non devono	

Operazione	Description	Competenze richieste
	necessariamente essere presenti nel bucket di destinazione dopo un'importazione. Lascia le altre impostazioni ai valori predefiniti. f. Importa: rivedi e scegli Importa quando sei pronto. Attendi il caricamento e l'importa zione dei dati. Per file di grandi dimensioni, Couchbase Capella supporta l'importazione da riga di comando tramite cURL. Puoi esplorare le opzioni di importazione in modo più dettagliato in Importa dati nella documentazione di Couchbase Capella.	

Testa e verifica la migrazione

Operazione	Description	Competenze richieste
Verifica la migrazione dei dati.	1. Nel piano di controllo di Couchbase Capella, scegli Clusters, quindi scegli il cluster di destinazione nell'elenco dei cluster. 2. Scegli la scheda Bucket per il tuo cluster di destinazi one. Verifica che il numero	Amministratore di Couchbase

Operazione	Description	Competenze richieste
	di elementi (documenti) nel bucket di destinazione corrisponda al numero di elementi nel bucket di origine. 3. Nel cluster di destinazione, nell'elenco a discesa Strumenti, scegli Documenti. Verifica che tutti i documenti siano stati migrati. 4. (Facoltativo) Dopo la migrazione di tutti i dati, è possibile interrompere la replica eliminandola. Per ulteriori informazioni, consulta Eliminare una replica nella documentazione di Couchbase.	
Verifica la migrazione dell'indice.	Nel piano di controllo di Couchbase Capella, nell'elenco a discesa Strumenti per il cluster di destinazione, scegli Indici. Verifica che gli indici siano migrati e creati.	Amministratore di Couchbase

Operazione	Description	Competenze richieste
Verifica i risultati della query.	1. Nel piano di controllo di Couchbase Capella, nell'elenco a discesa Strumenti per il cluster di destinazione, scegli Query Workbench. 2. Esegui una query N1QL di esempio o una query utilizzata nell'applicazione. Assicurati di ricevere gli stessi risultati di quando esegui la query nel cluster di Couchbase Server autogestito.	Amministratore di Couchbase
Verifica i risultati della ricerca in testo completo (applicabile se hai migrato gli indici FTS).	1. Nel piano di controllo di Couchbase Capella, nell'elenco a discesa Strumenti per il cluster di destinazione, scegli Ricerca a testo completo. 2. Seleziona un indice FTS scegliendone il nome. 3. Selezionare Search (Cerca). 4. Inserisci una query di ricerca di esempio e scegli Cerca. 5. Verifica che i risultati siano gli stessi di quando esegui la ricerca nel cluster autogestito.	Amministratore di Couchbase

Risorse correlate

Prepara la migrazione

- Inizia con la versione di prova gratuita di Couchbase Capella
- Requisiti del provider di cloud per Couchbase Capella
- Linee guida per il dimensionamento di Couchbase Capella

Migra i dati e gli indici

- Couchbase XDCR
- Documentazione Couchbase Shell

Couchbase Capella e supporto SLAs

- Accordi sui livelli di servizio di Couchbase Capella () SLAs
- Politica di supporto del servizio Couchbase Capella

Informazioni aggiuntive

Il codice seguente è un esempio di file di configurazione per Couchbase Shell.

```
Version = 1

[[clusters]]
identifier = "On-Prem-Cluster"
hostnames = ["<SELF_MANAGED_COUCHBASE_CLUSTER>"]
default-bucket = "travel-sample"
username = "<SELF_MANAGED_ADMIN>"
password = "<SELF_MANAGED_ADMIN_PWD>"
tls-cert-path = "/<ABSOLUTE_PATH_TO_SELF_MANAGED_ROOT_CERT>"
data-timeout = "2500ms"
connect-timeout = "7500ms"
query-timeout = "75s"

[[clusters]]
identifier = "Capella-Cluster"
hostnames = ["<COUCHBASE_CAPELLA_ENDPOINT>"]
default-bucket = "travel-sample"
username = "<CAPELLA_DATABASE_USER>"
```

```
password = "<CAPELLA_DATABASE_USER_PWD>"
tls-cert-path = "/<ABSOLUTE_PATH_TO_COUCHBASE_CAPELLA_ROOT_CERT>"
data-timeout = "2500ms"
connect-timeout = "7500ms"
query-timeout = "75s"
```

Prima di salvare il file di configurazione, utilizzate la tabella seguente per assicurarvi di aver aggiunto le informazioni sul cluster di origine e di destinazione.

<SELF_MANAGED_COUCHBASE_CLUSTER>	Usa l'indirizzo IP per il tuo cluster Couchbase Server autogestito.
<SELF_MANAGED_ADMIN>	Usa l'utente amministratore per il tuo cluster Couchbase Server autogestito.
<ABSOLUTE_PATH_TO_SELF_MANAGED_ROOT_CERT>	Usa il percorso assoluto del file di certificato root salvato per il tuo cluster Couchbase Server autogestito.
<COUCHBASE_CAPELLA_ENDPOINT>	Usa l'endpoint di connessione per il tuo cluster Couchbase Capella.
<CAPELLA_DATABASE_USER>	Usa l'utente del database per il tuo cluster Couchbase Capella.
<CAPELLA_DATABASE_USER_PWD>	Usa la password utente del database per il tuo cluster Couchbase Capella.
<ABSOLUTE_PATH_TO_COUCHBASE_CAPELLA_ROOT_CERT>	Usa il percorso assoluto del file di certificato root salvato per il tuo cluster Couchbase Capella.

Esegui la migrazione da IBM WebSphere Application Server ad Apache Tomcat su Amazon EC2

Neal Ardeljan e Afroz Khan, Amazon Web Services

Riepilogo

Questo modello illustra i passaggi per la migrazione da un sistema locale Red Hat Enterprise Linux (RHEL) 6.9 o successivo che esegue IBM WebSphere Application Server (WAS) a RHEL 8 con Apache Tomcat su un'istanza Amazon Elastic Compute Cloud (Amazon). EC2

Il modello può essere applicato alle seguenti versioni di origine e destinazione:

- WebSphere Application Server 7.x su Apache Tomcat 8 (con Java 7 o versione successiva)
- WebSphere Da Application Server 8.x ad Apache Tomcat 8 (con Java 7 o versione successiva)
- WebSphere Da Application Server 8.5.5.x ad Apache Tomcat 9 (con Java 8 o versione successiva)
- WebSphere Da Application Server 8.5.5.x ad Apache Tomcat 10 (con Java 8 o versione successiva)

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Codice sorgente Java, con i seguenti presupposti:
  - Utilizza la versione Java Development Kit (JDK) di Java 7 o successiva
  - Utilizza il framework Spring o Apache Struts
  - Non utilizza il framework Enterprise Java Beans (EJB) o qualsiasi altra funzionalità del WebSphere server non immediatamente disponibile per Tomcat
  - Utilizza principalmente servlet o Java Server Pages () JSPs
  - Utilizza i connettori Java Database Connectivity (JDBC) per connettersi ai database
- Source IBM WebSphere Application Server versione 7.x o successiva
- Target Apache Tomcat versione 8.5 o successiva

Architecture

Stack tecnologico di origine

- Un'applicazione web creata utilizzando il framework Apache Struts Model-View-Controller (MVC)
- Un'applicazione Web in esecuzione su IBM WebSphere Application Server versione 7.x o 8.x
- Un'applicazione web che utilizza un connettore LDAP (Lightweight Directory Access Protocol) per connettersi a una directory LDAP (iPlanet/eTrust)
- Un'applicazione che utilizza la connettività IBM Tivoli Access Manager (TAM) per aggiornare la password utente TAM (nella presente implementazione, le applicazioni utilizzano PD.jar)

Database locali

- Oracle Database 21c (21.0.0.0)
- Oracle Database 19c (19.0.0.0)
- Oracle Database 12c Release 2 (12.2.0.1)
- Oracle Database 12c Release 1 (12.1.0.2)

Stack tecnologico Target

- Apache Tomcat versione 8 (o successiva) in esecuzione su RHEL su un'istanza EC2
- Amazon Relational Database Service (Amazon RDS) per Oracle

Per ulteriori informazioni sulle versioni Oracle supportate da Amazon RDS, consulta il sito Web Amazon RDS for Oracle.

Architettura Target

Tools (Strumenti)

- Livello di applicazione: ricostruzione dell'applicazione Java in un file WAR.
- Livello database: backup e ripristino nativi di Oracle.
- Strumento di migrazione Apache Tomcat per Jakarta EE. Questo strumento utilizza un'applicazione web scritta per Java EE 8 che funziona su Apache Tomcat 9 e la converte automaticamente per l'esecuzione su Apache Tomcat 10, che implementa Jakarta EE 9.

Epiche

Pianifica la migrazione

Operazione	Description	Competenze richieste
Completa l'individuazione delle applicazioni, lo stato attuale e la base di riferimento delle prestazioni.		BA, responsabile della migrazione
Convalida le versioni del database di origine e di destinazione.		DBA
Identifica i requisiti hardware per l' EC2 istanza del server di destinazione.		DBA, SysAdmin
Identifica i requisiti di archiviazione (tipo e capacità di archiviazione).		DBA, SysAdmin
Scegli il tipo di EC2 istanza corretto in base alla capacità, alle funzionalità di archiviazione e alle funzionalità di rete.		DBA, SysAdmin
Identifica i requisiti di sicurezza dell'accesso alla rete per i database di origine e di destinazione.		DBA, SysAdmin
Identifica la strategia e gli strumenti di migrazione delle applicazioni.		DBA, responsabile della migrazione

Operazione	Description	Competenze richieste
Completa la progettazione e la guida alla migrazione per l'applicazione.		Build Lead, Migration Lead
Completa il runbook sulla migrazione delle applicazioni.		Build Lead, Cutover Lead, Testing Lead, Migration Lead

Configura l'infrastruttura

Operazione	Description	Competenze richieste
Crea un cloud privato virtuale (Virtual Private Cloud (VPC))		SysAdmin
Crea i gruppi di sicurezza.		SysAdmin
Configura e avvia Amazon RDS for Oracle.		DBA, SysAdmin

Migrazione dei dati

Operazione	Description	Competenze richieste
Crea o ottieni l'accesso agli endpoint per recuperare i file di backup del database.		DBA
Utilizza il motore di database nativo o uno strumento di terze parti per migrare oggetti e dati del database.	Per i dettagli, vedere «Migrazione di oggetti e dati del database» nella sezione Informazioni aggiuntive.	DBA

Migrare l'applicazione

Operazione	Description	Competenze richieste
Presenta la richiesta di modifica (CR) per la migrazione.		Cutover Lead
Ottieni l'approvazione CR per la migrazione.		Cutover Lead
Segui la strategia di migrazione delle applicazioni riportata nel runbook di migrazione delle applicazioni.	Per i dettagli, consulta «Configurazione del livello di applicazione» nella sezione Informazioni aggiuntive.	DBA, ingegnere addetto alla migrazione, proprietario dell'app
Aggiorna l'applicazione (se necessario).		DBA, ingegnere addetto alla migrazione, proprietario dell'app
Completa i test funzionali, non funzionali, di convalida dei dati, dello SLA e delle prestazioni.		Responsabile del test, proprietario dell'app, utenti dell'app

Tagliare

Operazione	Description	Competenze richieste
Ottenere l'approvazione dal proprietario dell'applicazione o dal titolare dell'attività.		Cutover Lead
Passa i client applicativi alla nuova infrastruttura.		DBA, ingegnere addetto alla migrazione, proprietario dell'app

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse AWS temporanee.		DBA, ingegnere addetto alla migrazione, SysAdmin
Rivedi e convalida i documenti del progetto.		Responsabile della migrazione
Raccogli parametri come il tempo necessario per la migrazione, la percentuale di attività manuali rispetto a quelle automatizzate e il risparmio sui costi.		Responsabile della migrazione
Chiudi il progetto e fornisci feedback.		Responsabile della migrazion e, proprietario dell'app

Risorse correlate

Riferimenti

- Documentazione di Apache Tomcat 10.0
- Documentazione di Apache Tomcat 9.0
- Documentazione di Apache Tomcat 8.0
- Guida all'installazione di Apache Tomcat 8.0
- Documentazione JNDI di Apache Tomcat
- Sito Web Amazon RDS per Oracle
- Prezzi di Amazon RDS
- Oracle e Amazon Web Services
- Oracle su Amazon RDS
- Implementazioni Multi-AZ di Amazon RDS

Tutorial e video

- Nozioni di base su Amazon RDS

Informazioni aggiuntive

Migrazione di oggetti e dati del database

Ad esempio, se utilizzi backup/restore utilità Oracle native:

1. Crea il backup Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) per i file di backup del database (opzionale).
2. Esegui il backup dei dati di Oracle DB nella cartella condivisa di rete.
3. Accedere al server di staging della migrazione per mappare la cartella di condivisione di rete.
4. Copia i dati dalla cartella di condivisione di rete al bucket S3.
5. Richiedi un'implementazione Amazon RDS Multi-AZ per Oracle.
6. Ripristina il backup del database locale su Amazon RDS for Oracle.

Configurazione del livello di applicazione

1. Installa Tomcat 8 (o 9/10) dal sito Web di Apache Tomcat.
2. Package dell'applicazione e delle librerie condivise in un file WAR.
3. Distribuisci il file WAR in Tomcat.
4. Monitora il registro di avvio su `Linux cat` tutte le librerie condivise mancanti da. WebSphere
5. Guarda il record iniziale di `Linux cat` qualsiasi estensione descrittrice di distribuzione WebSphere specifica.
6. Raccogli tutte le librerie Java dipendenti mancanti dal WebSphere server.
7. Modifica gli elementi del descrittore WebSphere di distribuzione specifici con equivalenti compatibili con Tomcat.
8. Ricostruisci il file WAR con le librerie Java dipendenti e i descrittori di distribuzione aggiornati.
9. Aggiorna la configurazione LDAP, la configurazione del database e le connessioni di test (vedi Realm Configuration HOW-TO e JNDI Datasource HOW-TO nella documentazione di Apache Tomcat).
10. Testa l'applicazione installata con il database Amazon RDS for Oracle ripristinato.
11. Crea un'Amazon Machine Image (AMI) per Linux dall' EC2 istanza.
12. Avvia l'architettura completa con il gruppo Application Load Balancer and Auto Scaling.

13 Aggiorna URLs (utilizzando la giunzione WebSeal) in modo che punti all'Application Load Balancer.

14 Aggiornare il database di gestione della configurazione (CMDB).

Esegui la migrazione da IBM WebSphere Application Server ad Apache Tomcat su Amazon EC2 con Auto Scaling

Kevin Yung e Afroz Khan, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni per la migrazione di un'applicazione Java da IBM WebSphere Application Server ad Apache Tomcat su un'istanza Amazon Elastic Compute Cloud (Amazon EC2) con Amazon Auto EC2 Scaling abilitato.

Utilizzando questo modello, è possibile ottenere:

- Una riduzione dei costi di licenza IBM
- Alta disponibilità grazie all'implementazione Multi-AZ
- Resilienza delle applicazioni migliorata con Amazon EC2 Auto Scaling

Prerequisiti e limitazioni

Prerequisiti

- Applicazioni Java (versione 7). x o 8. x) dovrebbe essere sviluppato in stack LAMP.
- Lo stato di destinazione è ospitare applicazioni Java su host Linux. Questo modello è stato implementato con successo in un ambiente Red Hat Enterprise Linux (RHEL) 7. Altre distribuzioni Linux possono seguire questo schema, ma è necessario fare riferimento alla configurazione della distribuzione Apache Tomcat.
- È necessario comprendere le dipendenze dell'applicazione Java.
- È necessario avere accesso al codice sorgente dell'applicazione Java per apportare modifiche.

Limitazioni e modifiche alla ripiattaforma

- È necessario conoscere i componenti dell'archivio aziendale (EAR) e verificare che tutte le librerie siano incluse nei file WAR dei componenti Web. È necessario configurare il plugin Apache Maven WAR e produrre artefatti di file WAR.
- Quando si utilizza Apache Tomcat 8, esiste un conflitto noto tra servlet-api.jar e i file jar incorporati nel pacchetto dell'applicazione. Per risolvere questo problema, eliminate servlet-api.jar dal pacchetto dell'applicazione.

- È necessario configurare WEB-INF/Resources che si trovano nel classpath della configurazione di Apache Tomcat. Per impostazione predefinita, le librerie JAR non vengono caricate nella directory. In alternativa, puoi distribuire tutte le risorse insrc/main/resources.
- Verifica la presenza di eventuali root contestuali codificate all'interno dell'applicazione Java e aggiorna la nuova radice contestuale di Apache Tomcat.
- Per impostare le opzioni di runtime JVM, è possibile creare il file di configurazione setenv.sh nella cartella bin di Apache Tomcat; ad esempio, JAVA_OPTS, JAVA_HOME, ecc.
- L'autenticazione è configurata a livello di contenitore ed è configurata come realm nelle configurazioni di Apache Tomcat. L'autenticazione viene stabilita per uno dei tre ambiti seguenti:
  - JDBC Database Realm cerca gli utenti in un database relazionale a cui accede il driver JDBC.
  - DataSource Database Realm cerca gli utenti in un database a cui accede JNDI.
  - JNDI Directory Realm cerca gli utenti nella directory LDAP (Lightweight Directory Access Protocol) a cui accede il provider JNDI. Le ricerche richiedono:
    - Dettagli della connessione LDAP: base di ricerca utente, filtro di ricerca, base di ruoli, filtro di ruolo
    - La chiave JNDI Directory Realm: si connette a LDAP, autentica gli utenti e recupera tutti i gruppi di cui un utente è membro
- Autorizzazione: nel caso di un contenitore con un'autorizzazione basata sui ruoli che controlla i vincoli di autorizzazione in web.xml, le risorse web devono essere definite e confrontate con i ruoli definiti nei vincoli. Se LDAP non dispone della mappatura dei ruoli di gruppo, è necessario impostare l'attributo < > in web.xml per ottenere la mappatura dei ruoli di gruppo. security-role-ref Per vedere un esempio di documento di configurazione, consulta la documentazione Oracle.
- Connessione al database: crea una definizione di risorsa in Apache Tomcat con un URL dell'endpoint Amazon Relational Database Service (Amazon RDS) e dettagli di connessione. Aggiorna il codice dell'applicazione in modo che faccia riferimento a utilizzando la ricerca JNDI DataSource . Una connessione DB esistente definita in non WebSphere funzionerebbe, poiché utilizza i nomi WebSphere JNDI di. È possibile aggiungere una <resource-ref>voce in web.xml con il nome JNDI e la definizione del DataSource tipo. Per visualizzare un documento di configurazione di esempio, consultate la documentazione di Apache Tomcat.
- Registrazione: per impostazione predefinita, Apache Tomcat accede alla console o a un file di registro. È possibile abilitare la traccia a livello di realm aggiornando logging.properties (vedi Registrazione in Tomcat). Se utilizzi Apache Log4j per aggiungere log a un file, devi scaricare tomcat-juli e aggiungerlo al classpath.

- Gestione delle sessioni: se utilizzi IBM WebSeal per il bilanciamento del carico delle applicazioni e la gestione delle sessioni, non è richiesta alcuna modifica. Se utilizzi un Application Load Balancer o Network Load Balancer su AWS per sostituire il componente IBM WebSeal, devi configurare la gestione delle sessioni utilizzando un'istanza ElastiCache Amazon con un cluster Memcached e configurare Apache Tomcat per utilizzare la gestione delle sessioni open source.
- Se utilizzi il forward proxy IBM WebSEAL, devi configurare un nuovo Network Load Balancer su AWS. Utilizza le configurazioni di giunzione IPs fornite dal Network Load Balancer per le configurazioni di giunzione WebSEAL.
- Configurazione SSL: si consiglia di utilizzare Secure Sockets Layer (SSL) per le comunicazioni. end-to-end Per configurare una configurazione del server SSL in Apache Tomcat, segui le istruzioni nella documentazione di Apache Tomcat.

Architecture

Stack di tecnologia di origine

- Server di applicazioni IBM WebSphere

Stack tecnologico Target

- L'architettura utilizza Elastic Load Balancing (versione 2). Se utilizzi IBM WebSeal per la gestione e il bilanciamento del carico di Identify, puoi selezionare un Network Load Balancer su AWS da integrare con il reverse proxy IBM WebSeal.
- Le applicazioni Java vengono distribuite su un server applicativo Apache Tomcat, che viene eseguito su un' EC2 istanza in un gruppo Amazon Auto Scaling EC2 . Puoi impostare una politica di scalabilità basata su CloudWatch parametri di Amazon come l'utilizzo della CPU.
- Se stai ritirando l'uso di IBM WebSeal per il bilanciamento del carico, puoi utilizzare Amazon for Memcached ElastiCache per la gestione delle sessioni.
- Per il database di back-end, puoi implementare High Availability (Multi-AZ) per Amazon RDS e selezionare un tipo di motore di database.

Architettura Target

Tools (Strumenti)

- AWS CloudFormation
- Interfaccia a riga di comando AWS (AWS CLI)
- Apache Tomcat (versione 7). x o 8. x)
- RHEL 7 o Centos 7
- Implementazione di Amazon RDS Multi-AZ
- Amazon ElastiCache per Memcached (opzionale)

Epiche

Configura il VPC

Operazione	Description	Competenze richieste
Crea un cloud privato virtuale (Virtual Private Cloud (VPC))		
Creare sottoreti.		
Se necessario, create tabelle di routing.		
Crea liste di controllo degli accessi alla rete (ACLs).		
Configura AWS Direct Connect o una connessione VPN aziendale.		

Ripiattaforma l'applicazione

Operazione	Description	Competenze richieste
Rifattorizza la configura zione Maven di build dell'appl		

Operazione	Description	Competenze richieste
icazione per generare gli artefatti WAR.		
Rifattorizza le fonti di dati sulle dipendenze delle applicazioni in Apache Tomcat.		
Rifattorizza i codici sorgente dell'applicazione per utilizzar e i nomi JNDI in Apache Tomcat.		
Distribuisci gli artefatti WAR in Apache Tomcat.		
Completa le convalide e i test delle applicazioni.		

Configura la rete

Operazione	Description	Competenze richieste
Configura il firewall aziendale per consentire la connessione ai servizi di dipendenza.		
Configura il firewall aziendale per consentire all'utente finale l'accesso a Elastic Load Balancing on AWS.		

Crea l'infrastruttura applicativa

Operazione	Description	Competenze richieste
Crea e distribuisci l'applica zione su un' EC2 istanza.		
Crea un cluster Amazon ElastiCache for Memcached per la gestione delle sessioni.		
Crea un'istanza Amazon RDS Multi-AZ per il database di backend.		
Crea certificati SSL e importali in AWS Certificate Manager (ACM).		
Installa i certificati SSL sui sistemi di bilanciamento del carico.		
Installa i certificati SSL per i server Apache Tomcat.		
Convalide e test completi delle applicazioni.		

Tagliare

Operazione	Description	Competenze richieste
Chiudere l'infrastruttura esistente.		
Ripristina il database dalla produzione ad Amazon RDS.		

Operazione	Description	Competenze richieste
Riduci l'applicazione apportando modifiche al DNS.		

Risorse correlate

Riferimenti

- Documentazione di Apache Tomcat 7.0
- Guida all'installazione di Apache Tomcat 7.0
- Documentazione JNDI di Apache Tomcat
- Implementazioni Multi-AZ di Amazon RDS
- Amazon ElastiCache per Memcached

Tutorial e video

- Nozioni di base su Amazon RDS

Esegui la migrazione di un'applicazione.NET da Microsoft Azure App Service ad AWS Elastic Beanstalk

Raghavender Madamshitti, Amazon Web Services

Riepilogo

Questo modello descrive come migrare un'applicazione Web.NET ospitata su Microsoft Azure App Service su AWS Elastic Beanstalk. Esistono due modi per migrare le applicazioni su Elastic Beanstalk:

- Usa AWS Toolkit for Visual Studio: questo plugin per l'IDE di Microsoft Visual Studio offre il modo più semplice e diretto per distribuire applicazioni.NET personalizzate in AWS. Puoi utilizzare questo approccio per distribuire il codice.NET direttamente in AWS e creare risorse di supporto, come Amazon Relational Database Service (Amazon RDS) per i database SQL Server, direttamente da Visual Studio.
- Carica e distribuisci su Elastic Beanstalk: ogni servizio app di Azure include un servizio in background chiamato Kudu, utile per acquisire dump di memoria e log di distribuzione, visualizzare i parametri di configurazione e accedere ai pacchetti di distribuzione. Puoi usare la console Kudu per accedere ai contenuti del Servizio app di Azure, estrarre il pacchetto di distribuzione e quindi caricare il pacchetto su Elastic Beanstalk usando l'opzione di caricamento e distribuzione nella console Elastic Beanstalk.

Questo modello descrive il secondo approccio (caricamento dell'applicazione su Elastic Beanstalk tramite Kudu). Il modello utilizza anche i seguenti servizi AWS: AWS Elastic Beanstalk, Amazon Virtual Private Cloud (Amazon VPC), Amazon, Amazon Elastic Compute Cloud (CloudWatchAmazon) Auto Scaling, EC2 Amazon Simple Storage Service (Amazon S3) e Amazon Route 53.

L'applicazione Web.NET viene distribuita su AWS Elastic Beanstalk, che viene eseguito in un Amazon EC2 Auto Scaling Group. Puoi impostare una politica di scalabilità basata su CloudWatch parametri di Amazon come l'utilizzo della CPU. Per un database, puoi utilizzare Amazon RDS in un ambiente Multi-AZ o Amazon DynamoDB, a seconda dell'applicazione e dei requisiti aziendali.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo

- Un'applicazione Web.NET in esecuzione nel servizio app di Azure
- Autorizzazione a usare la console Azure App Service Kudu

Versioni del prodotto

- .NET Core (x64) 1.0.1, 2.0.0 o versione successiva oppure .NET Framework 4.x, 3.5 (consulta .NET nella cronologia della piattaforma Windows Server)
- Internet Information Services (IIS) versione 8.0 o successiva, in esecuzione su Windows Server 2012 o versione successiva
- .NET 2.0 o 4.0 Runtime.

Architecture

Stack di tecnologia di origine

- Applicazione sviluppata usando .NET Framework 3.5 o versione successiva oppure .NET Core 1.0.1, 2.0.0 o versione successiva e ospitata nel Servizio app di Azure (app Web o app API)

Stack tecnologico Target

- AWS Elastic Beanstalk in esecuzione in un gruppo Amazon EC2 Auto Scaling

Architettura di migrazione

Workflow di distribuzione

Tools (Strumenti)

Strumenti

- .NET Core o.NET Framework
- C#
- IIS
- Console Kudu

Servizi e funzionalità AWS

- AWS Elastic Beanstalk — Elastic Beanstalk easy-to-use è un servizio per la distribuzione e il ridimensionamento di applicazioni web.NET. Elastic Beanstalk gestisce automaticamente il provisioning della capacità, il bilanciamento del carico e la scalabilità automatica.
- Gruppo Amazon EC2 Auto Scaling: Elastic Beanstalk include un gruppo Auto Scaling che gestisce le istanze Amazon nell'ambiente. EC2 In un ambiente con singola istanza, il gruppo Auto Scaling garantisce che esista sempre un'istanza in esecuzione. In un ambiente con carico bilanciato, puoi configurare il gruppo con una gamma di istanze da eseguire e Amazon Auto Scaling EC2 aggiunge o rimuove istanze secondo necessità, in base al carico.
- Elastic Load Balancing: quando abiliti il bilanciamento del carico in AWS Elastic Beanstalk, crea un sistema di bilanciamento del carico che distribuisce il traffico tra le istanze dell'ambiente. EC2
- Amazon CloudWatch: Elastic Beanstalk CloudWatch utilizza automaticamente Amazon per fornire informazioni sulle risorse dell'applicazione e dell'ambiente. Amazon CloudWatch supporta metriche standard, metriche personalizzate e allarmi.
- Amazon Route 53 — Amazon Route 53 è un servizio Web DNS (Domain Name System) cloud altamente disponibile e scalabile. Puoi utilizzare i record di alias Route 53 per mappare nomi di dominio personalizzati in ambienti AWS Elastic Beanstalk.

Epiche

Configurazione di un VPC

Operazione	Description	Competenze richieste
Configura un cloud privato virtuale (VPC).	Nel tuo account AWS, crea un VPC con le informazioni richieste.	Amministratore di sistema
Crea sottoreti.	Crea due o più sottoreti nel tuo VPC.	Amministratore di sistema
Crea una tabella di rotte.	Crea una tabella di percorsi, in base alle tue esigenze.	Amministratore di sistema

Configura Elastic Beanstalk

Operazione	Description	Competenze richieste
Accedi alla console Azure App Service Kudu.	Accedi a Kudu tramite il portale di Azure accedendo alla dashboard del Servizio app e scegliendo Strumenti avanzati, Vai. In alternativa, puoi modificare l'URL del servizio app di Azure come segue:. `https://<appservicename>.scm.azurewebsites.net`	Sviluppatore di app, amministratore di sistema
Scarica il pacchetto di distribuzione da Kudu.	Passa a Windows PowerShell scegliendo l'DebugConsoleopzione. Si aprirà la console Kudo. Vai alla `wwwroot` cartella e scaricala. Verrà scaricato il pacchetto di distribuzione del servizio app di Azure come file zip. Per un esempio, vedi l'allegato.	Sviluppatore di app, amministratore di sistema
Crea un pacchetto per Elastic Beanstalk.	Decomprimi il pacchetto di distribuzione scaricato dal Servizio app di Azure. Crea un file JSON chiamato `aws-windows-deployment-manifest.json` (questo file è richiesto solo per le applicazioni.NET Core). Crea un file zip che `aws-windows-deployment-manifest.json` includa il file del pacchetto di distribuzione	Sviluppatore di app, amministratore di sistema

Operazione	Description	Competenze richieste
	del servizio app di Azure. Per un esempio, vedi l'allegato.	
Crea una nuova applicazione Elastic Beanstalk.	Apri la console Elastic Beanstalk. Scegli un'applic azione esistente o crea una nuova applicazione.	Sviluppatore di app, amministr atore di sistema
Creazione dell'ambiente	Nel menu Azioni della console Elastic Beanstalk, scegli Crea ambiente. Seleziona l'ambiente del server Web e la piattaforma .NET/IIS. Per il codice dell'applicazione, scegli Upload. Carica il file zip che hai preparato per Elastic Beanstalk, quindi scegli Crea ambiente.	Sviluppatore di app, amministr atore di sistema
Configura Amazon CloudWatc h.	Per impostazione predefinita, il CloudWatch monitoraggio di base è abilitato. Se desideri modificare la configurazione, nella procedura guidata Elastic Beanstalk, scegli l'applica zione pubblicata, quindi scegli Monitoring.	Amministratore di sistema
Verifica che il pacchetto di distribuzione sia in Amazon S3.	Una volta creato l'ambient e applicativo, puoi trovare il pacchetto di distribuzione nel bucket S3.	Sviluppatore di app, amministr atore di sistema

Operazione	Description	Competenze richieste
Testare l'applicazione.	Una volta creato l'ambient e, utilizza l'URL fornito nella console Elastic Beanstalk per testare l'applicazione.	Amministratore di sistema

Risorse correlate

- Concetti di AWS Elastic Beanstack (documentazione di Elastic Beanstalk)
- Guida introduttiva a .NET su Elastic Beanstalk (documentazione di Elastic Beanstalk)
- GitHubConsole Kudu ()
- Usare «Kudu» per gestire le app Web di Azure (articolo di GS Lab)
- Distribuzioni personalizzate di ASP.NET Core Elastic Beanstalk (guida per l'utente di AWS Toolkit for Visual Studio)
- Documentazione su Elastic Load Balancing
- Piattaforme supportate da AWS Elastic Beanstalk (documentazione Elastic Beanstalk)
- Distribuzione di un'applicazione Web su AWS (articolo C# Corner)
- Ridimensionamento delle dimensioni del gruppo Auto Scaling (documentazione Amazon EC2)
- Alta disponibilità (Multi-AZ) per Amazon RDS (documentazione Amazon RDS)

Informazioni aggiuntive

Note

- Se stai migrando un database SQL Server locale o di Azure su Amazon RDS, devi aggiornare anche i dettagli della connessione al database.
- A scopo di test, è allegata un'applicazione demo di esempio.

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Esegui la migrazione da Oracle WebLogic ad Apache Tomcat (ToMee) su Amazon ECS

Anya Epishcheva e Harshad Gohil, Amazon Web Services

Riepilogo

Questo modello illustra i passaggi per la migrazione di un sistema Oracle Solaris SPARC locale che esegue Oracle verso un'installazione basata su container Docker che esegue Apache Tomee (Apache Tomcat con supporto WebLogic per container aggiunto) con Amazon Elastic Container Service (Amazon ECS).

Per informazioni sulla migrazione dei database associati alle applicazioni che stai migrando da Oracle a Tomcat, consulta i modelli di migrazione dei database in questo catalogo. WebLogic

Best practice

I passaggi per la migrazione delle applicazioni Web Java e Java Enterprise Edition (Java EE) variano a seconda del numero di risorse specifiche del contenitore utilizzate dall'applicazione. Le applicazioni basate su Spring sono in genere più facili da migrare, perché hanno un numero limitato di dipendenze dal contenitore di distribuzione. Al contrario, le applicazioni Java EE che utilizzano Enterprise JavaBeans (EJBs) e risorse container gestite come pool di thread, Java Authentication and Authorization Service (JAAS) e Container-Managed Persistence (CMP) richiedono uno sforzo maggiore.

Le applicazioni sviluppate per Oracle Application Server utilizzano spesso la suite Oracle Identity Management. I clienti che migrano a server applicativi open source spesso scelgono di reimplementare la gestione delle identità e degli accessi utilizzando la federazione basata su SAML. Altri utilizzano Oracle HTTP Server Webgate nei casi in cui la migrazione dalla suite Oracle Identity Management non è un'opzione.

Le applicazioni Web Java e Java EE sono ottimi candidati per la distribuzione su servizi AWS basati su Docker, come AWS Fargate e Amazon ECS. I clienti scelgono spesso un'immagine Docker con la versione più recente del server delle applicazioni di destinazione (come ToMee) e il Java Development Kit (JDK) preinstallato. Installano le loro applicazioni sull'immagine Docker di base, la pubblicano nel registro Amazon Elastic Container Registry (Amazon ECR) e la utilizzano per la distribuzione scalabile delle loro applicazioni su AWS Fargate o Amazon ECS.

Idealmente, la distribuzione delle applicazioni è elastica; vale a dire, il numero di istanze dell'applicazione aumenta o diminuisce, a seconda del traffico o del carico di lavoro. Ciò significa che le istanze delle applicazioni devono essere online o chiuse per adattare la capacità alla domanda.

Quando sposti un'applicazione Java in AWS, valuta la possibilità di renderla stateless. Questo è un principio architettonico chiave di AWS Well-Architected Framework che consentirà la scalabilità orizzontale utilizzando la containerizzazione. Ad esempio, la maggior parte delle applicazioni Web basate su Java archivia localmente le informazioni sulla sessione utente. Per sopravvivere alla chiusura dell'istanza dell'applicazione dovuta al ridimensionamento automatico in Amazon Elastic Compute Cloud (Amazon EC2) o per altri motivi, le informazioni sulla sessione utente devono essere archiviate a livello globale in modo che gli utenti delle applicazioni Web possano continuare a lavorare senza interruzioni e in modo trasparente senza riconnettersi o riaccedere a un'applicazione Web. Esistono diverse opzioni architettoniche per questo approccio, tra cui Amazon ElastiCache for Redis o l'archiviazione dello stato della sessione in un database globale. I server di applicazioni come TomEE dispongono di plug-in che consentono l'archiviazione e la gestione delle sessioni tramite Redis, database e altri archivi di dati globali.

Utilizza uno strumento di registrazione e debug comune e centralizzato, facilmente integrabile con Amazon e AWS X-Ray. CloudWatch La migrazione offre l'opportunità di migliorare le funzionalità del ciclo di vita delle applicazioni. Ad esempio, potresti voler automatizzare il processo di creazione in modo che le modifiche possano essere apportate facilmente utilizzando una pipeline di integrazione e distribuzione continua (CI/CD). Ciò potrebbe richiedere modifiche all'applicazione in modo che possa essere distribuita senza tempi di inattività.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Codice sorgente Java e JDK
- Applicazione sorgente creata con Oracle WebLogic
- Soluzione definita per la gestione delle identità e degli accessi (SAML o Oracle Webgate)
- Soluzione definita per la gestione delle sessioni delle applicazioni (spostamento like-for-like o con Amazon o creazione dello stato dell'applicazione ElastiCache, se necessario)
- Capire se il team deve rifattorizzare le librerie specifiche di J2EE per la portabilità su Apache ToMee (vedi Java EE 7 Implementation Status sul sito web di Apache)
- Immagine ToMee rafforzata in base ai requisiti di sicurezza

- Immagine del contenitore con destinazione ToMee preinstallata
- Correzione dell'applicazione concordata e implementata se necessario (ad esempio, registrazione, debug, build, autenticazione)

Versioni del prodotto

- Oracle WebLogic OC4 J, 9i, 10g
- Tomcat 7 (con Java 1.6 o successivo)

Architecture

Stack tecnologico di origine

- Applicazione Web creata con Oracle WebLogic
- Applicazione Web che utilizza l'autenticazione Oracle Webgate o SAML
- Applicazioni Web connesse a Oracle Database versione 10g e successive

Stack tecnologico Target

- ToMee (Apache Tomcat con supporto aggiuntivo per container) in esecuzione su Amazon ECS (vedi anche Deploying Java Web Applications and Java Microservices on Amazon ECS)
- Amazon Relational Database Service (Amazon RDS) per Oracle; per le versioni Oracle supportate da Amazon RDS, consulta Amazon RDS per Oracle

Architettura di Target

Tools (Strumenti)

Per funzionare su TomEE, un'applicazione Java deve essere ricostruita in un file.war. In alcuni casi, potrebbero essere necessarie modifiche all'applicazione per far funzionare l'applicazione su TomEE; è necessario verificare che le opzioni di configurazione e le proprietà dell'ambiente necessarie siano definite correttamente.

Inoltre, le ricerche JNDI (Java Naming and Directory Interface) e gli spazi dei nomi JavaServer Pages (JSP) devono essere definiti correttamente. Valuta la possibilità di controllare i nomi dei

file utilizzati dall'applicazione per evitare collisioni di denominazione con le librerie T integrate. Ad esempio, persistence.xml è un nome di file utilizzato dal framework Apache OpenJPA (fornito in bundle con OpenEJB in TomEE) per scopi di configurazione. Il file persistence.xml in PUI contiene le dichiarazioni dei bean del framework Spring.

TomEE versione 7.0.3 e successive (Tomcat 8.5.7 e successive) restituisce una risposta HTTP 400 (richiesta errata) per raw (non codificato) con caratteri speciali. URLs La risposta del server viene visualizzata come una pagina vuota per l'utente finale. Le versioni precedenti di ToMee e Tomcat consentivano l'uso di determinati caratteri speciali non codificati in URLs; tuttavia, è considerato pericoloso, come indicato nel sito Web CVE-2016-6816. Per risolvere il problema di codifica degli URL, i dati URLs passati direttamente al browser JavaScript devono essere codificati con il metodo encodeURI () anziché essere utilizzati come stringhe non elaborate.

Dopo aver distribuito il file.war in TomEE, monitora il log di avvio su Linux cat per eventuali librerie condivise mancanti ed estensioni specifiche di Oracle per aggiungere i componenti mancanti dalle librerie Tomcat.

Procedura generale

- Configura l'applicazione su TomEE.
- Identifica e riconfigura i file e le risorse di configurazione specifici del server delle applicazioni dal formato di origine a quello di destinazione.
- Identifica e riconfigura le risorse JNDI.
- Adatta lo spazio dei nomi e le ricerche EJB al formato richiesto dal server delle applicazioni di destinazione (se applicabile).
- Riconfigurate i ruoli di sicurezza e le mappature principali specifici del contenitore dell'applicazione JAAS (se applicabile).
- Package dell'applicazione e delle librerie condivise in un file.war.
- Distribuisci il file.war in TomEE utilizzando il contenitore Docker fornito.
- Monitora il registro di avvio per identificare eventuali estensioni mancanti della libreria condivisa e del descrittore di distribuzione. Se ne vengono trovate, tornate alla prima attività.
- Testa l'applicazione installata con il database Amazon RDS ripristinato.
- Avvia l'architettura completa con un sistema di bilanciamento del carico e un cluster Amazon ECS seguendo le istruzioni in Deploy Docker Containers.
- Aggiorna il punto URLs di bilanciamento del carico.

- Aggiornare il database di gestione della configurazione (CMDB).

Poemi epici

Pianifica la migrazione

Operazione	Description	Competenze richieste
Esegui l'individuazione delle applicazioni (impronta dello stato attuale e baseline delle prestazioni).		BA, responsabile della migrazione
Convalida le versioni e i motori del database di origine e destinazione.		DBA
Convalida il design dell'appl icazione di origine e di destinazione (gestione dell'identità e della sessione).		DBA, ingegnere addetto alla migrazione, proprietario dell'app
Identifica i requisiti hardware e di storage per l'istanza del server di destinazione.		DBA, SysAdmin
Scegli il tipo di istanza corretto in base alla capacità, alle funzionalità di archiviazione e alle funzionalità di rete.		DBA, SysAdmin
Identifica i requisiti di sicurezza dell'accesso alla rete per i database di origine e di destinazione.		DBA, SysAdmin

Operazione	Description	Competenze richieste
Identifica la strategia e gli strumenti di migrazione delle applicazioni.		DBA, responsabile della migrazione
Completa la progettazione e la guida alla migrazione per l'applicazione.		Build Lead, Migration Lead
Completa il runbook sulla migrazione delle applicazioni.		Build Lead, Cutover Lead, Testing Lead, Migration Lead

Configura l'infrastruttura

Operazione	Description	Competenze richieste
Crea un cloud privato virtuale (Virtual Private Cloud (VPC))		SysAdmin
Crea gruppi di sicurezza.		SysAdmin
Configura e avvia l'istanza database di Amazon RDS.		DBA, SysAdmin
Configura la distribuzione di Amazon ECS.		SysAdmin
Package della tua applicazione come immagine Docker.		SysAdmin
Invia l'immagine al registro Amazon ECR (o salta questo passaggio e inviala al cluster Amazon ECS).		SysAdmin
Configura la definizione delle attività per l'applicazione e le		SysAdmin

Operazione	Description	Competenze richieste
opzioni del servizio Amazon ECS.		
Configura il cluster, rivedi le impostazioni di sicurezza e imposta i ruoli di AWS Identity and Access Management (IAM).		SysAdmin
Avvia la configurazione ed esegui i test in base al runbook di migrazione delle applicazioni.		SysAdmin

Migrazione dei dati

Operazione	Description	Competenze richieste
Ottieni l'autorizzazione del tuo team di garanzia della sicurezza per spostare i dati di produzione in AWS.		DBA, ingegnere addetto alla migrazione, proprietario dell'app
Crea e ottieni l'accesso agli endpoint per recuperare i file di backup del database.		DBA
Utilizza il motore di database nativo o strumenti di terze parti per migrare oggetti e dati del database.		DBA
Esegui i test necessari dal runbook di migrazione delle		DBA, ingegnere addetto alla migrazione, proprietario dell'app

Operazione	Description	Competenze richieste
applicazioni per confermare la corretta migrazione dei dati.		

Esegui la migrazione dell'applicazione

Operazione	Description	Competenze richieste
Crea una richiesta di modifica (CR) per la migrazione.		Cutover Lead
Ottieni l'approvazione CR per la migrazione.		Cutover Lead
Segui la strategia di migrazione delle applicazioni riportata nell'Application Migration Runbook.		DBA, ingegnere addetto alla migrazione, proprietario dell'app
Aggiorna l'applicazione (se necessario).		DBA, ingegnere addetto alla migrazione, proprietario dell'app
Test completi, funzionali, non funzionali, di convalida dei dati, degli SLA e delle prestazioni.		Responsabile del test, proprietario dell'app, utenti dell'app

Tagliare

Operazione	Description	Competenze richieste
Ottenere l'approvazione dall'applicazione o dal titolare dell'attività.		Cutover Lead

Operazione	Description	Competenze richieste
Esegui un esercizio tematico da tavolo per illustrare tutti i passaggi del cutover runbook.		DBA, ingegnere addetto alla migrazione, proprietario dell'app
Passa i client applicativi alla nuova infrastruttura.		DBA, ingegnere addetto alla migrazione, proprietario dell'app

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse AWS temporanee.		DBA, ingegnere addetto alla migrazione, SysAdmin
Rivedi e convalida i documenti del progetto.		Responsabile della migrazione
Raccogli le metriche in tempo utile per la migrazione, la percentuale di utilizzo manuale rispetto agli strumenti, i risparmi sui costi, ecc.		Responsabile della migrazione
Chiudi il progetto e fornisci feedback.		Responsabile della migrazione, proprietario dell'app

Risorse correlate

Riferimenti

- Documentazione di Apache Tomcat 7.0
- Guida all'installazione di Apache Tomcat 7.0
- Documentazione JNDI di Apache Tomcat
- Documentazione Apache TomEE

- Amazon RDS per Oracle
- Prezzi di Amazon RDS
- Oracle e AWS
- Documentazione Oracle su Amazon RDS
- Implementazioni Amazon RDS Multi-AZ
- Guida introduttiva ad Amazon ECS
- Nozioni di base su Amazon RDS

Tutorial e video

- Best practice per l'esecuzione di database Oracle su Amazon RDS (presentazione re:Invent 2018)

Esegui la migrazione di un database Oracle da Amazon EC2 ad Amazon RDS for Oracle utilizzando AWS DMS

Chethan Gangadharaiah e Brian Motzer, Amazon Web Services

Riepilogo

Questo modello descrive i passaggi per la migrazione di un database Oracle da Amazon Elastic Compute Cloud (Amazon EC2) ad Amazon Relational Database Service (Amazon RDS) per Oracle utilizzando AWS Database Migration Service (AWS DMS). Il modello utilizza anche Oracle SQL Developer o SQL *Plus per connettersi all'istanza Oracle DB e include un CloudFormation modello AWS che automatizza alcune attività.

La migrazione ad Amazon RDS for Oracle ti consente di concentrarti sulla tua attività e sulle tue applicazioni, mentre Amazon RDS si occupa delle attività di amministrazione del database come il provisioning dei database, il backup e il ripristino, le patch di sicurezza, gli aggiornamenti delle versioni e la gestione dello storage.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un'Amazon Machine Image (AMI) per Oracle Database su Amazon EC2

Versioni del prodotto

- AWS DMS supporta le versioni Oracle 11g (versione 11.2.0.3.v1 e successive), 12c e 18c per i database di istanze Amazon RDS per le edizioni Enterprise, Standard, Standard One e Standard Two. Per le informazioni più recenti sulle versioni supportate, consulta Using an Oracle Database as a Target for AWS DMS nella documentazione AWS. (I CloudFormation modelli AWS allegati utilizzano la versione 12c di Oracle come database di origine.)
- Oracle SQL Developer 4.0.3

Architecture

Architettura di origine

- Database Oracle su Amazon EC2

Architettura Target

- Amazon RDS per Oracle

Architettura di migrazione

Tools (Strumenti)

- AWS DMS: AWS Database Migration Service (AWS DMS) ti aiuta a migrare i database su AWS in modo rapido e sicuro. Supporta migrazioni sia omogenee che eterogenee. Per informazioni sulle versioni e le edizioni del database Oracle supportate, consulta Using an Oracle Database as a Source for AWS DMS e Using an Oracle Database as a Target for AWS DMS nella documentazione AWS.
- Oracle SQL Developer o SQL *Plus: questi strumenti consentono di connettersi all'istanza DB Amazon RDS for Oracle.

Epiche

Configura il tuo database di destinazione

Operazione	Description	Competenze richieste
Crea un'istanza database Amazon RDS for Oracle.	Accedere alla Console di gestione AWS e aprire la console Amazon RDS all'indirizzo https://console.aws.amazon.com/rds/. Crea un'istanza DB Oracle selezionando il motore, il modello, l'impostazione delle credenziali del database, il tipo di istanza, lo storage, le impostazioni Multi-AZ, il cloud privato virtuale (VPC) e la configurazione, le credenziali di accesso e le impostazi	Developer

Operazione	Description	Competenze richieste
	oni aggiuntive per il database Oracle. Per istruzioni, visualizz a i collegamenti nella sezione «Risorse correlate». Oppure utilizza il CloudFormation modello AWS (Create_R DS.yaml) nell'allegato per creare l'istanza DB Amazon RDS for Oracle.	
Connect ad Amazon RDS e concedi i privilegi all'utente Oracle.	Modifica il gruppo di sicurezza per aprire le porte appropria te per la connessione dalla macchina locale e dall'ista nza di replica AWS DMS. Quando configuri la connettiv ità, assicurati che l'opzione «Accessibile pubblicamente» sia selezionata in modo da poterti connettere al database dall'esterno del VPC. Connettit i ad Amazon RDS con Oracle SQL Developer o SQL *Plus utilizzando le credenziali di accesso, crea un utente AWS DMS e fornisci i privilegi richiesti all'utente AWS DMS per modificare il database.	Developer

Configura il gruppo di sicurezza dell'istanza di origine EC2

Operazione	Description	Competenze richieste
Controlla se il database Oracle è attivo e funzionante.	Usa Secure Shell (SSH) per connetterti all' EC2 istanza	Developer

Operazione	Description	Competenze richieste
	e prova a connetterti al database Oracle utilizzando SQL *Plus.	
Modifica il gruppo di sicurezza.	Modifica il gruppo di sicurezza dell' EC2 istanza per aprire le porte appropriate, in modo da poterti connettere dal tuo computer locale e dall'istanza di replica AWS DMS.	Developer

Configura AWS DMS

Operazione	Description	Competenze richieste
Creare un'istanza di replica di AWS DMS.	In AWS DMS, crea un'istanz a di replica nello stesso VPC dell'istanza DB Amazon RDS for Oracle. Specificare il nome e la descrizione dell'istanza di replica, scegliere la classe di istanza e la versione del motore di replica (utilizzare l'impostazione predefinita), scegliere il VPC in cui è stata creata l'istanza database Amazon RDS, impostare le impostazioni Multi-AZ se necessario, allocare lo storage, specificare la zona di disponibilità e configura re impostazioni aggiuntive. In alternativa, puoi utilizzar e il CloudFormation modello AWS (DMS.yaml) nell'alle	DBA

Operazione	Description	Competenze richieste
	gato per implementare questo passaggio.	
Connect agli endpoint del database di origine e di destinazione.	Crea gli endpoint del database di origine e di destinazione specificando l'identificatore dell'endpoint, il motore, il server, la porta, le credenzia li di accesso e gli attributi di connessione aggiuntiv i. Per il server di origine, utilizza il DNS pubblico dell' EC2 istanza che ospita il database Oracle. Per il server di destinazione, utilizza l'endpoint di Amazon RDS for Oracle. Esegui un test per verificare che le connessioni di origine e di destinazione funzionino. In alternativa, puoi utilizzare il CloudFormation modello AWS (DMS.yaml) nell'allegato per implementare questo passaggio.	DBA

Operazione	Description	Competenze richieste
Crea un'attività AWS DMS.	Crea un'attività AWS DMS per migrare i dati dall'endpoint di origine all'endpoint di destinazione, per configurare la replica tra l'endpoint di origine e quello di destinazione o entrambi. Quando crei il task AWS DMS, specifica l'istanza di replica, l'endpoint di origine, l'endpoint di destinazione, il tipo di migrazione (solo dati, solo replica o entrambi), la mappatura delle tabelle e il filtro. Esegui l'attività AWS DMS, monitora l'attività, controlla le statistiche della tabella e controlla i log in Amazon. CloudWatch In alternativa, puoi utilizzare il CloudFormation modello AWS (DMS.yaml) nell'allegato per implementare questo passaggio.	DBA

Risorse correlate

- Creazione di un'istanza database Amazon RDS
- Connessione a un'istanza database che esegua il motore di database di Oracle
- Documentazione AWS DMS
- Procedure dettagliate di AWS DMS Step-by-Step
- Migrazione dei database Oracle sul cloud AWS

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Esegui la migrazione di un database Oracle locale ad Amazon OpenSearch Service utilizzando Logstash

Aditya Goteti, Amazon Web Services

Riepilogo

Questo modello descrive come spostare i dati da un database Oracle locale ad Amazon OpenSearch Service utilizzando Logstash. Include considerazioni sull'architettura e alcuni set di competenze e consigli richiesti. I dati possono provenire da una singola tabella o da più tabelle in cui sarà necessario eseguire una ricerca completa.

OpenSearch Il servizio può essere configurato all'interno di un cloud privato virtuale (VPC) oppure può essere collocato pubblicamente con restrizioni basate su IP. Questo modello descrive uno scenario in cui il OpenSearch servizio è configurato all'interno di un VPC. Logstash viene utilizzato per raccogliere i dati dal database Oracle, analizzarli in formato JSON e quindi inserire i dati in Service. OpenSearch

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Java 8 (richiesto da Logstash 6.4.3)
- Connettività tra i server di database locali e le istanze Amazon Elastic Compute Cloud EC2 (Amazon) in un VPC, stabilita utilizzando AWS Virtual Private Network (AWS VPN)
- Una query per recuperare i dati richiesti da inviare a Service dal database OpenSearch
- Driver JDBC (Oracle Java Database Connectivity)

Limitazioni

- Logstash non è in grado di identificare i record eliminati definitivamente dal database

Versioni del prodotto

- Database Oracle 12c
- OpenSearch Servizio 6.3
- Logstash 6.4.3

Architecture

Stack tecnologico di origine

- Database Oracle locale
- VPN AWS locale

Stack tecnologico Target

- VPC
- EC2 istanza
- OpenSearch servizio
- Logstash
- NAT Gateway (per gli aggiornamenti del sistema operativo sulle EC2 istanze e per l'installazione di Java 8, Logstash e plugin)

Architettura di migrazione dei dati

Tools (Strumenti)

- Logstash 6.4.3
- Plugin di input JDBC (download e ulteriori informazioni)
- Plugin di output Logstash (_es) logstash-output-amazon
- Driver Oracle JDBC

Epiche

Pianifica la migrazione

Operazione	Description	Competenze richieste
Identifica la dimensione dei dati di origine.	La dimensione dei dati di origine è uno dei parametri utilizzati per determinare il	DBA, sviluppatore di database

Operazione	Description	Competenze richieste
	numero di shard da configura re in un indice.	
Analizza i tipi di dati di ogni colonna e i dati corrispondenti.	OpenSearch Il servizio mappa dinamicamente il tipo di dati quando nel documento viene trovato un campo mai visto in precedenza. Se esistono tipi o formati di dati specifici (ad esempio campi data) che devono essere dichiarati in modo esplicito, identifica i campi e definisci la mappatura per tali campi durante la creazione dell'indice.	Proprietario dell'app, sviluppat ore, sviluppatore di database
Determina se ci sono colonne con chiavi primarie o uniche.	Per evitare la duplicazi one dei record in Amazon OpenSearch Service durante gli aggiornamenti o gli inserimenti, devi configurare l'`document_id` impostazi one nella sezione di output del `amazon_es` plug-in (ad esempio, `document_id => "%{customer_id}"` `customer_id` dov'è una chiave primaria).	Proprietario dell'app, sviluppat ore

Operazione	Description	Competenze richieste
Analizza il numero e la frequenza dei nuovi record aggiunti; controlla la frequenza con cui i record vengono eliminati.	Questa attività è necessaria per comprendere il tasso di crescita dei dati di origine. Se i dati vengono letti in modo intensivo e gli inserimenti sono rari, è possibile disporre di un unico indice. Se i nuovi record vengono inseriti frequente mente e non vi sono eliminazi oni, la dimensione dello shard può facilmente superare la dimensione massima consiglia ta di 50 GB. In questo caso, puoi creare dinamicamente un indice configurando i modelli di indice in Logstash e nel codice a cui puoi accedervi utilizzan do un alias.	Proprietario dell'app, sviluppat ore
Determina quante repliche sono necessarie.		Proprietario dell'app, sviluppat ore
Determina il numero di shard da configurare nell'indice.		Proprietario dell'app, sviluppat ore
Identifica i tipi di istanza per i nodi master dedicati, i nodi di dati e l' EC2 istanza.	Per ulteriori informazioni, consulta la sezione Risorse correlate.	Proprietario dell'app, sviluppat ore
Determina il numero di nodi master e nodi dati dedicati richiesti.	Per ulteriori informazioni, consulta la sezione Risorse correlate.	

Migrazione dei dati

Operazione	Description	Competenze richieste
Avvia un' EC2 istanza.	Avvia un' EC2 istanza all'interno del VPC a cui è connessa AWS VPN.	Costrutti Amazon VPC, AWS VPN
Installa Logstash sull'istanza. EC2		Developer
Installa i plugin Logstash.	Installa i plugin Logstash richiesti e. `jdbc-input` `logstash-output-amazon_es`	Developer
Configura Logstash.	Crea il keystore Logstash per archiviare informazioni sensibili come le chiavi di AWS Secrets Manager e le credenziali del database, quindi inserisci i riferimenti in un file di configurazione Logstash.	Developer
Configura la coda di lettere morte e la coda persistente.	Per impostazione predefinita, quando Logstash rileva un evento che non può elaborare perché i dati contengono un errore di mappatura o qualche altro problema, la pipeline Logstash si blocca o elimina l'evento non riuscito. Per proteggersi dalla perdita di dati in questa situazione, puoi configurare Logstash in modo che scriva gli eventi non riusciti in una coda di lettere	Developer

Operazione	Description	Competenze richieste
	morte anziché eliminarli. Per proteggersi dalla perdita di dati durante un'interruzione anomala, Logstash dispone di una funzione di coda persisten te che memorizzerà la coda dei messaggi su disco. Le code persistenti garantisc ono la durabilità dei dati in Logstash.	
Crea il dominio Amazon OpenSearch Service.	Crea il dominio Amazon OpenSearch Service con una policy di accesso che non richieda la firma di richieste con credenziali AWS Identity and Access Management (IAM). Il dominio Amazon OpenSearch Service deve essere creato all'interno dello stesso VPC. Dovresti anche selezionare i tipi di istanza e impostare il numero di nodi dedicati e master in base all'analisi.	Developer
Configura i log di Amazon OpenSearch Service richiesti.	Per ulteriori informazioni, consulta la documentazione del OpenSearch servizio.	
Crea l'indice.		Developer

Operazione	Description	Competenze richieste
Avvia Logstash.	Esegui Logstash come servizio in background. Logstash esegue la query SQL configurata, estrae i dati, li converte in formato JSON e li invia al servizio. OpenSearch Per il caricamento iniziale, non configurare lo scheduler nel file di configurazione di Logstash.	Developer

Operazione	Description	Competenze richieste
Controlla i documenti.	Controlla il numero di documenti nell'indice e se tutti i documenti sono presenti nel database di origine. Durante il caricamento iniziale, vengono aggiunti all'indice e utilizzati per arrestare Logstash. Modifica la configurazione di Logstash per aggiunger e uno scheduler che venga eseguito a intervalli fissi in base ai requisiti del client e riavvia Logstash. Logstash selezionerà solo i record che sono stati aggiornat i o aggiunti dopo l'ultima esecuzione e il timestamp dell'ultima esecuzione viene archiviato nel file configura to con la proprietà nel file di configurazione di Logstash. `last_run_metadata_ path => "/usr/sha re/logstash/.logst ash_jdbc_last_run"`	Developer

Risorse correlate

- Allarmi consigliati CloudWatch
- Nodi master dedicati OpenSearch di Amazon Service
- Dimensionamento dei domini di OpenSearch servizio Amazon
- Documentazione Logstash
- Plugin di input JDBC

- Plugin di output Logstash
- Sito web Amazon OpenSearch Service

Esegui la migrazione di un database Oracle locale ad Amazon RDS for Oracle

Baji Shaik e Pavan Pusuluri, Amazon Web Services

Riepilogo

Questo modello descrive i passaggi per la migrazione dei database Oracle locali ad Amazon Relational Database Service (Amazon RDS) per Oracle. Come parte del processo di migrazione, crei un piano di migrazione e consideri i fattori importanti dell'infrastruttura di database di destinazione in base al database di origine. È possibile scegliere una delle due opzioni di migrazione in base ai requisiti aziendali e al caso d'uso:

- AWS Database Migration Service (AWS DMS): puoi utilizzare AWS DMS per migrare i database nel cloud AWS in modo rapido e sicuro. Il database di origine rimane pienamente operativo durante la migrazione, il che riduce al minimo i tempi di inattività delle applicazioni che si basano sul database. Puoi ridurre i tempi di migrazione utilizzando AWS DMS per creare un'attività che registri le modifiche in corso dopo aver completato una migrazione iniziale a pieno carico tramite un processo chiamato change data capture (CDC).
- Strumenti Oracle nativi: puoi migrare i database utilizzando strumenti Oracle nativi, come Oracle e Data Pump Export e Data Pump Import withOracle for CDC. GoldenGate È inoltre possibile utilizzare strumenti Oracle nativi come l'utilità Export originale e l'utilità Import originale per ridurre il tempo di caricamento completo.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database Oracle locale
- Un'istanza di database Amazon RDS Oracle (DB)

Limitazioni

- Limite di dimensione del database: 64 TB

Versioni del prodotto

- Versioni Oracle 11g (versioni 11.2.0.3.v1 e successive) e fino a 12.2 e 18c. Per l'elenco più recente delle versioni ed edizioni supportate, consulta Amazon RDS for Oracle nella documentazione AWS. Per le versioni Oracle supportate da AWS DMS, consulta Using an Oracle database as a source for AWS DMS nella documentazione di AWS DMS.

Architecture

Stack tecnologico di origine

- Database Oracle locali

Stack tecnologico Target

- Amazon RDS per Oracle

Architettura di origine e destinazione

Il diagramma seguente mostra come migrare un database Oracle locale ad Amazon RDS for Oracle utilizzando AWS DMS.

Il diagramma mostra il flusso di lavoro seguente:

1. Crea o usa un utente del database esistente, concedi le autorizzazioni AWS DMS richieste a quell'utente, attiva la modalità ARCHIVELOG e quindi configura la registrazione supplementare.
2. Configura il gateway Internet tra la rete locale e la rete AWS.
3. Configura gli endpoint di origine e destinazione per AWS DMS.
4. Configura le attività di replica di AWS DMS per migrare i dati dal database di origine al database di destinazione.
5. Completa le attività successive alla migrazione sul database di destinazione.

Il diagramma seguente mostra come migrare un database Oracle locale ad Amazon RDS for Oracle utilizzando strumenti Oracle nativi.

Il diagramma mostra il flusso di lavoro seguente:

1. Crea o utilizza un utente del database esistente e concedi le autorizzazioni necessarie per eseguire il backup del database Oracle utilizzando le utilità Oracle Export (`exp`) e Import (`imp`).
2. Configura il gateway Internet tra la rete locale e la rete AWS.
3. Configura il client Oracle sull'host Bastion per utilizzare il database di backup.
4. Carica il database di backup in un bucket Amazon Simple Storage Service (Amazon S3).
5. Ripristina il backup del database da Amazon S3 su un database Amazon RDS for Oracle.
6. Configura Oracle GoldenGate per CDC.
7. Completa le attività successive alla migrazione sul database di destinazione.

Tools (Strumenti)

- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali.
- Gli strumenti nativi di Oracle ti aiutano a eseguire una migrazione omogenea. È possibile utilizzare Oracle Data Pump per migrare i dati tra i database di origine e di destinazione. Questo modello utilizza Oracle Data Pump per eseguire il caricamento completo dal database di origine al database di destinazione.
- Oracle GoldenGate consente di eseguire la replica logica tra due o più database. Questo modello viene utilizzato GoldenGate per replicare le modifiche delta dopo il caricamento iniziale utilizzando Oracle Data Pump.

Epiche

Pianifica la migrazione

Operazione	Description	Competenze richieste
Crea documenti di progetto e registra i dettagli del database.	1. Documenta gli obiettivi di migrazione, i requisiti di migrazione, i principali stakeholder del progetto, le tappe fondamentali del progetto, le scadenze del progetto, le metriche chiave, i rischi di migrazion	DBA

Operazione	Description	Competenze richieste
	e e i piani di mitigazione dei rischi. 2. Documenta le informazioni critiche sul tuo database di origine, tra cui RAM, IOPS e. CPUs Successiv amente utilizzerai queste informazioni per determina re l'istanza DB di destinazi one appropriata. 3. Convalida le versioni dei database di origine e di destinazione.	

Operazione	Description	Competenze richieste
Identifica i requisiti di archiviazione.	Identifica e documenta i tuoi requisiti di archiviazione, tra cui: 1. Calcola lo storage allocato per l'istanza DB di origine. 2. Raccogli le metriche di crescita storiche dall'istanza DB di origine. 3. Prevedi le crescite future per l'istanza DB di destinazione. **Note** Per i volumi SSD General Purpose (gp2), ottieni tre IOPS per 1 GB di storage. Alloca lo storage calcolando il numero totale di IOPS di lettura e scrittura sul database di origine.	DBA, SysAdmin

Operazione	Description	Competenze richieste
Scegli il tipo di istanza corretto in base ai requisiti di elaborazione.	1. Determina i requisiti di calcolo dell'istanza DB di destinazione. 2. Identifica i problemi di prestazioni. 3. Considerate i fattori per determinare il tipo di istanza appropriato: • Utilizzo della CPU dell'istanza DB di origine • IOPS (lettura e scrittura) per l'istanza DB di origine • Impronta di memoria sull'istanza DB di origine	SysAdmin
Identifica i requisiti di sicurezza dell'accesso alla rete.	1. Identifica e documenta i requisiti di sicurezza dell'accesso alla rete per i database di origine e di destinazione. 2. Configura i gruppi di sicurezza appropriati per consentire all'applicazione di comunicare con il database.	DBA, SysAdmin

Operazione	Description	Competenze richieste
Identifica la strategia di migrazione delle applicazioni.	1. Determina e documenta la strategia di transizione alla migrazione. 2. Determina e documenta il Recovery Time Objective (RTO) e il Recovery Point Objective (RPO) dell'appl icazione, quindi pianifica il cutover di conseguenza.	DBA, SysAdmin proprietario dell'app
Identifica i rischi legati alla migrazione.	Valuta i rischi e le mitigazio ni specifici della migrazione del database e dei documenti. Esempio: • Identifica le tabelle senza registrazione ed evidenzia il rischio di perdita dei dati in caso di ripristino. • Estrai gli utenti e i privilegi del database di origine ed evidenzia i conflitti con i privilegi di Amazon RDS. • Controlla il registro degli avvisi per eventuali errori e avvisi specifici di Oracle. • Identifica le funzionalità supportate e non supportat e dell'istanza DB di destinazi one. • Esamina le funzionalità obsolete del motore della versione DB di destinazione.	DBA

Configura l'infrastruttura

Operazione	Description	Competenze richieste
Crea un VPC.	Crea un nuovo Amazon Virtual Private Cloud (Amazon VPC) per l'istanza DB di destinazione.	SysAdmin
Crea gruppi di sicurezza.	Crea un gruppo di sicurezza nel tuo nuovo VPC per consentire le connessioni in entrata all'istanza DB.	SysAdmin
Crea un'istanza database Amazon RDS for Oracle.	Crea l'istanza DB di destinazione con il nuovo VPC e il nuovo gruppo di sicurezza, quindi avvia l'istanza.	SysAdmin

Opzione 1: utilizza strumenti nativi Oracle o di terze parti per migrare i dati

Operazione	Description	Competenze richieste
Preparare il database di origine.	1. Crea una directory Data Pump o usane una esistente. 2. Crea un utente di migrazione e concedi le autorizzazioni per eseguire l'estrazione di Data Pump. 3. Estrai ruoli, utenti e tablespace dal database di origine come script SQL. 4. Trasferisci il dump di Data Pump estratto nella	DBA, SysAdmin

Operazione	Description	Competenze richieste
	directory dell'istanza DB di destinazione. `data pump`	

Operazione	Description	Competenze richieste
Preparare il database di destinazione.	1. Verifica che tutte le opzioni del database (ad esempio testo e Java) siano installate o abilitate sull'istanza DB Amazon RDS for Oracle di destinazione. 2. Crea una directory Data Pump o usane una esistente. 3. Crea un utente di migrazione e concedi le autorizzazioni per eseguire l'importazione di Data Pump. 4. Crea i tablespace, gli utenti e i ruoli richiesti sull'istanza DB di destinazione. 5. Importa il dump di esportazione di Data Pump trasferito nel database di destinazione. 6. Crea tutti gli indici esclusi durante l'importazione o la creazione dell'oggetto. 7. Crea tutti i vincoli esclusi durante l'importazione. 8. Convalida o ricompila oggetti non validi. 9. Ricostruisci gli indici non validi. 10. Convalida il conteggio degli oggetti del database tra il database di origine e quello di destinazione.	DBA, SysAdmin

Operazione	Description	Competenze richieste
	11. Risolve eventuali discrepanze rilevate tra i conteggi degli oggetti.	

Opzione 2: usa AWS DMS per migrare i dati

Operazione	Description	Competenze richieste
Preparare i dati.	1. Pulisci i dati nel database di origine. 2. Crea un'istanza di replica. 3. Crea un endpoint di origine e un endpoint di destinazione. 4. Identifica il numero di tabelle e oggetti da migrare.	DBA
Migrare i dati.	1. Elimina i vincoli e i trigger di chiave esterna nel database di destinazione. 2. Elimina gli indici secondari sul database di destinazione. 3. Configura le impostazioni delle attività a pieno carico di AWS DMS dal database di origine al database di destinazione. 4. Abilita le chiavi esterne. 5. Abilita AWS DMS CDC per replicare le modifiche in corso. 6. Abilita i trigger.	DBA

Operazione	Description	Competenze richieste
	7. Aggiorna le sequenze. 8. Convalida i dati di origine e di destinazione.	

Passa al database di destinazione

Operazione	Description	Competenze richieste
Passa i client applicativi alla nuova infrastruttura.	1. Interrompi tutti i servizi applicativi e le connessioni client che puntano a Oracle. 2. Esegui le attività di AWS DMS. 3. Imposta un'attività di rollback (ad esempio, inverti il CDC dal database Amazon RDS al database Oracle locale). 4. Convalida i dati. 5. Avvia i servizi applicativi sul nuovo database di destinazione configurando Amazon Route 53 sulla nuova istanza DB Amazon RDS for Oracle. 6. Aggiungi il CloudWatch monitoraggio di Amazon alla tua nuova istanza DB Amazon RDS for Oracle.	DBA, proprietario dell' SysAdminapp
Implementa il tuo piano di rollback.	1. Interrompi tutti i servizi applicativi che puntano	DBA, proprietario dell'app

Operazione	Description	Competenze richieste
	all'istanza DB di Amazon RDS for Oracle. 2. Ripristina le modifiche al database Oracle locale di origine utilizzando un task AWS DMS. 3. Interrompi l'esecuzione delle attività AWS DMS dal database Oracle locale al database Amazon RDS for Oracle. 4. Configura nuovamente le applicazioni sul database Oracle di origine. 5. Conferma che la distribuz ione del rollback sia completa.	

Chiudi il progetto di migrazione

Operazione	Description	Competenze richieste
Eliminare le risorse.	Chiudi o rimuovi le risorse AWS temporanee, come l'istanza di replica AWS DMS e il bucket S3.	DBA, SysAdmin
Rivedi i documenti del progetto.	Esamina i documenti e gli obiettivi di pianificazione della migrazione, quindi conferma di aver completato tutti i passaggi di migrazione richiesti.	DBA SysAdmin, proprietario dell'app

Operazione	Description	Competenze richieste
Raccogli le metriche.	Registra i principali parametri di migrazione, tra cui il tempo impiegato per completare la migrazione, la percentuale di attività manuali rispetto a quelle basate su strumenti , i risparmi sui costi e altre metriche pertinenti.	DBA, proprietario dell'app SysAdmin
Chiudi il progetto.	Chiudi il progetto di migrazion e e raccogli il feedback sull'iniz iativa.	DBA SysAdmin, proprietario dell'app

Risorse correlate

Riferimenti

- Migrazione dei database Oracle al cloud AWS (AWS Prescriptive Guidance)
- AWS Database Migration Service (documentazione AWS DMS)
- Prezzi di Amazon RDS (documentazione Amazon RDS)

Tutorial e video

- Guida introduttiva ad AWS Database Migration Service (documentazione AWS DMS)
- Risorse Amazon RDS (documentazione Amazon RDS)
- AWS Database Migration Service (DMS) (YouTube)

Esegui la migrazione di un database Oracle locale ad Amazon RDS for Oracle utilizzando Oracle Data Pump

Mohan Annam e Brian Motzer, Amazon Web Services

Riepilogo

Questo modello descrive come migrare un database Oracle da un data center locale a un'istanza Amazon Relational Database Service (Amazon RDS) per Oracle DB utilizzando Oracle Data Pump.

Il modello prevede la creazione di un file di dump dei dati dal database di origine, l'archiviazione del file in un bucket Amazon Simple Storage Service (Amazon S3) e il ripristino dei dati su un'istanza DB Amazon RDS for Oracle. Questo modello è utile in caso di limitazioni nell'utilizzo di AWS Database Migration Service (AWS DMS) per la migrazione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Le autorizzazioni necessarie per creare ruoli in AWS Identity and Access Management (IAM) e per un caricamento multiparte di Amazon S3
- Le autorizzazioni necessarie per esportare i dati dal database di origine
- AWS Command Line Interface (AWS CLI) installata e configurata

Versioni del prodotto

- Oracle Data Pump è disponibile solo per Oracle Database 10g Release 1 (10.1) e versioni successive.

Architecture

Stack tecnologico di origine

- Database Oracle locali

Stack tecnologico Target

- Amazon RDS per Oracle

- Client SQL (Oracle SQL Developer)
- Un bucket S3

Architettura di origine e destinazione

Tools (Strumenti)

Servizi AWS

- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle. In questo modello, IAM viene utilizzato per creare i ruoli e le policy necessari per la migrazione dei dati da Amazon S3 ad Amazon RDS for Oracle.
- Amazon Relational Database Service (Amazon RDS) per Oracle ti aiuta a configurare, gestire e scalare un database relazionale Oracle nel cloud AWS.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri strumenti

- Oracle Data Pump ti aiuta a spostare dati e metadati da un database all'altro a velocità elevate. In questo modello, Oracle Data Pump viene utilizzato per esportare il file di dump dei dati (.dmp) sul server Oracle e per importarlo in Amazon RDS for Oracle. Per ulteriori informazioni, consulta Importazione di dati in Oracle su Amazon RDS nella documentazione di Amazon RDS.
- Oracle SQL Developer è un ambiente di sviluppo integrato che semplifica lo sviluppo e la gestione dei database Oracle in implementazioni tradizionali e basate su cloud. Interagisce sia con il database Oracle locale che con Amazon RDS for Oracle per eseguire i comandi SQL necessari per l'esportazione e l'importazione dei dati.

Epiche

Creare un bucket S3

Operazione	Description	Competenze richieste
Crea il bucket.	Per creare il bucket S3, segui le istruzioni nella documenta zione AWS.	Amministratore di sistema AWS

Crea il ruolo IAM e assegna le politiche

Operazione	Description	Competenze richieste
Configura le autorizzazioni IAM.	Per configurare le autorizza zioni, segui le istruzioni nella documentazione di AWS.	Amministratore di sistema AWS

Crea l'istanza database Amazon RDS for Oracle di destinazione e associa il ruolo di integrazione di Amazon S3

Operazione	Description	Competenze richieste
Crea l'istanza database Amazon RDS for Oracle di destinazione.	Per creare l'istanza Amazon RDS for Oracle, segui le istruzioni nella documenta zione AWS.	Amministratore di sistema AWS
Associa il ruolo all'istanza DB.	Per associare il ruolo all'istan za, segui le istruzioni nella documentazione AWS.	DBA

Crea l'utente del database sul database di destinazione

Operazione	Description	Competenze richieste

Creare l’utente.	Connect al database Amazon RDS for Oracle di destinazione da Oracle SQL Developer o SQL*Plus ed esegui il seguente comando SQL per creare l'utente in cui importare lo schema.
	`create user SAMPLE_SCHEMA identified by <PASSWORD>;`
	`grant create session, resource to <USER NAME>;`
	`alter user <USER NAME> quota 100M on users;`
		DBA

Crea il file di esportazione dal database Oracle di origine

Operazione	Description	Competenze richieste

Crea un file di dump dei dati.	Per creare un file di dump denominato `sample.dmp` nella `DATA_PUMP_DIR` directory per l'esportazione dell'`SAMPLE_SCHEMA` utente, utilizzare lo script seguente.
	`DECLARE`
	`  hdnl NUMBER;`
	`BEGIN`
	`  hdnl := dbms_datapump.open(operation => 'EXPORT',`
		DBA

Operazione	Description	Competenze richieste

	job_mode => 'SCHEMA',
	job_name => NULL);
	dbms_datapump.add_file(handle => hdnl,
	filename => 'sample.dmp',
	directory => 'DATA_PUMP_DIR',
	filetype => dbms_datapump.ku$_file_type_dump_file);
	dbms_datapump.add_file(handle => hdnl,
	filename => 'export.log',
	directory => 'DATA_PUMP_DIR',
	filetype => dbms_datapump.ku$_file_type_log_file);
	dbms_datapump.metadata_filter(hdnl, 'SCHEMA_EXPR', 'IN (''SAMPLE_SCHEMA'')');
	dbms_datapump.start_job(hdnl);
	END;	

Operazione	Description	Competenze richieste
	`/` Controlla i dettagli dell'esportazione esaminando il `export.log` file nella tua directory locale`DATA_PUMP_DIR` .	

Carica il file di dump nel bucket S3

Operazione	Description	Competenze richieste
Carica il file di dump dei dati dall'origine al bucket S3.	Utilizzando AWS CLI, esegui il seguente comando. `aws s3 cp sample.dmp s3://<bucket_created_epic_1>/`	DBA

Scarica il file di esportazione dal bucket S3 all'istanza RDS

Operazione	Description	Competenze richieste
Scarica il file di dump dei dati su Amazon RDS	Per copiare il file di dump `sample.dmp` dal bucket S3 al database Amazon RDS for Oracle, esegui il seguente comando SQL. In questo esempio, il `sample.dmp` file viene scaricato dal `my-s3-integration1` bucket S3 nella directory Oracle. `DATA_PUMP_DIR` Assicurati di disporre di spazio su disco	Amministratore di sistema AWS

Operazione	Description	Competenze richieste

	sufficiente all'istanza RDS per ospitare sia il database che il file di esportazione.
	
	`-- If you want to download all the files in the S3 bucket remove the p_s3_prefix line.`
	
	`SELECT rdsadmin.rdsadmin_s3_tasks.download_from_s3( p_bucket_name => 'my-s3-integration', p_s3_prefix => 'sample.dmp', p_directory_name => 'DATA_PUMP_DIR') AS TASK_ID FROM DUAL;`
	
	Il comando precedente restituisce un ID di attività. Per verificare lo stato del download esaminando i dati nell'ID dell'attività, esegui il comando seguente.
	
	`SELECT text FROM table(rdsadmin.rds_file_util.read_text_file('BDUMP','dbtask-<task_id>.log'));`	

Operazione	Description	Competenze richieste
	Per visualizzare i file nella `DATA_PUMP_DIR` directory, esegui il comando seguente. `SELECT filename, type,filesize/1024 /1024 size_megs ,to_char(mtime,'DD -MON-YY HH24:MI:SS') timestamp FROM TABLE(rdsadmin.rds _file_util.listdir (p_directory => upper('DATA_PUMP_D IR'))) order by 4;`	

Importa il file di dump nel database di destinazione

Operazione	Description	Competenze richieste
Ripristina lo schema e i dati su Amazon RDS.	Per importare il file di dump nello schema del `sample_sc hema` database, esegui il seguente comando SQL da SQL Developer o SQL*Plus. `DECLARE hdnl NUMBER; BEGIN hdnl := DBMS_DATA PUMP.OPEN( operation => 'IMPORT', job_mode => 'SCHEMA', job_name= >null); DBMS_DATAPUMP.ADD_ FILE( handle => hdnl,`	DBA

Operazione	Description	Competenze richieste

	filename => 'sample.dmp', directory => 'DATA_PUMP_DIR', filetype => dbms_datapump.ku$_file_type_dump_file);
	
	DBMS_DATAPUMP.ADD_FILE(handle => hdnl, filename => 'import.log', directory => 'DATA_PUMP_DIR', filetype => dbms_datapump.ku$_file_type_log_file);
	
	DBMS_DATAPUMP.METADATA_FILTER(hdnl,'SCHEMA_EXPR','IN (''SAMPLE_SCHEMA'')');
	
	DBMS_DATAPUMP.START_JOB(hdnl);
	
	END;
	/
	
	Per visualizzare il file di registro dell'importazione, esegui il comando seguente.
	
	SELECT text FROM table(rdsadmin.rds_file_util.read_text_file('DATA_PUMP_DIR','import.log'));	

Rimuovi il file di dump dalla directory DATA_PUMP_DIR

Operazione	Description	Competenze richieste

Elenca e pulisci i file di esportazione.	Elenca e rimuovi i file di esportazione nella `DATA_PUMP_DIR` directory, esegui i seguenti comandi.
	`-- List the files`
	`SELECT filename, type,filesize/1024 /1024 size_megs ,to_char(mtime,'DD -MON-YY HH24:MI:S S') timestamp FROM TABLE(rdsadmin.rds _file_util.listdir (p_directory => upper('DATA_PUMP_D IR'))) order by 4;`
	`-- Remove the files`
	`EXEC UTL_FILE. FREMOVE('DATA_PUMP _DIR','sample.dmp');`
	`EXEC UTL_FILE.FREMOVE(' DATA_PUMP_DIR','im port.log');`
		Amministratore di sistema AWS

Risorse correlate

- Integrazione con Amazon S3
- Crea un'istanza DB
- Importazione di dati in Oracle su Amazon RDS
- Documentazione Amazon S3
- Documentazione IAM

- Documentazione Amazon RDS
- Documentazione Oracle Data Pump
- Oracle SQL Developer

Esegui la migrazione da PostgreSQL su Amazon RDS per PostgreSQL utilizzando EC2 pglogical

Rajesh Madiwale, Amazon Web Services

Riepilogo

Questo modello descrive i passaggi per la migrazione di un database PostgreSQL (versione 9.5 e successive) da Amazon Elastic Compute Cloud (Amazon) ad Amazon Relational Database EC2 Service (Amazon RDS) per PostgreSQL utilizzando l'estensione pglogica PostgreSQL. Amazon RDS ora supporta l'estensione pglogical per PostgreSQL versione 10.

Prerequisiti e limitazioni

Prerequisiti

- Scegli il tipo giusto di istanza Amazon RDS. Per ulteriori informazioni, consulta la sezione Tipi di istanze Amazon RDS.
- Assicurati che le versioni di origine e di destinazione di PostgreSQL siano le stesse.
- Installa e integra l'estensione pglogical con PostgreSQL su Amazon. EC2

Versioni del prodotto

- PostgreSQL versione 10 e successive su Amazon RDS, con le funzionalità supportate su Amazon RDS (vedi PostgreSQL su Amazon RDS nella documentazione AWS). Questo modello è stato testato migrando PostgreSQL 9.5 alla versione 10 di PostgreSQL su Amazon RDS, ma si applica anche alle versioni successive di PostgreSQL su Amazon RDS.

Architecture

Architettura di migrazione dei dati

Tools (Strumenti)

- estensione pglogica
- Utilità native di PostgreSQL: pg_dump e pg_restore

Epiche

Migra i dati utilizzando l'estensione pglogical

Operazione	Description	Competenze richieste
Crea un'istanza database Amazon RDS PostgreSQL.	Configura un'istanza DB PostgreSQL in Amazon RDS. Per istruzioni, consulta la documentazione di Amazon RDS for PostgreSQL.	DBA
Ottieni un dump dello schema dal database PostgreSQL di origine e ripristinalo nel database PostgreSQL di destinazione.	1. Utilizza l'utilità pg_dump con l'opzione per generare un file di schema dal `-s` database di origine. 2. Utilizzate l'utilità psql con l'`-f`opzione per caricare lo schema nel database di destinazione.	DBA
Attiva la decodifica logica.	Nel gruppo di parametri Amazon RDS DB, imposta il parametro `rds.logical_replication` statico su 1. Per istruzioni, consulta la documentazione di Amazon RDS.	DBA
Crea l'estensione pglogical sui database di origine e di destinazione.	1. Crea l'`pglogical` estensione sul database PostgreSQL di origine: `psql -h <amazon-ec2-endpoint> -d target-dbname -U target-dbuser -c "create`	DBA

Operazione	Description	Competenze richieste

	`extension pglogical ;"`
	2. Crea l'`pglogical` estensione sul database PostgreSQL di destinazione:
	`psql -h <amazon-rds-endpoint> -d source-dbname -U source-dbuser -c "create extension pglogical ;"`	

Crea un editore sul database PostgreSQL di origine.	Per creare un editore, esegui: `psql -d dbname -p 5432 <<EOF SELECT pglogical.create_node( node_name := 'provider1', dsn := 'host=<ec2-endpoint> port=5432 dbname=source-dbname user=source-dbuser' ); EOF`	DBA

Operazione	Description	Competenze richieste
Crea un set di replica, aggiungi tabelle e sequenze.	Per creare un set di replica sul database PostgreSQL di origine e aggiungere tabelle e sequenze al set di replica, esegui: `psql -d dbname -p 5432 <<EOF SELECT pglogical .replication_set_add_all_tables('default', '{public}'::text[],synchronize_data := true); EOF`	DBA
Crea un abbonato.	Per creare un sottoscrittore sul database PostgreSQL di destinazione, esegui: `psql -h <rds-endpoint> -d target-dbname -U target-dbuser <<EOF SELECT pglogical .create_node( node_name := 'subscriber1', dsn := 'host=<rds-endpoint> port=5432 dbname=target-dbname password=postgres user=target-dbuser' ); EOF`	DBA

Operazione	Description	Competenze richieste
Crea un abbonamento.	Per creare un abbonamento sul database PostgreSQL di destinazione, esegui: `psql -h <rds-endpoint> -d target -U postgres <<EOF SELECT pglogical .create_subscription( subscription_name := 'subscription1', replication_sets := array['default'], provider_dsn := 'host=<ec2-endpoint> port=5432 dbname=<source-dbname> password=<password> user=source-dbuser' );`	DBA

Convalida i tuoi dati

Operazione	Description	Competenze richieste
Controlla i database di origine e di destinazione.	Controlla i database di origine e di destinazione per confermare che i dati vengano replicati correttamente. È possibile eseguire la convalida di base utilizzando le `select count(1)` tabelle di origine e destinazione.	DBA

Risorse correlate

- Amazon RDS
- Replica logica per PostgreSQL su Amazon RDS (documentazione Amazon RDS)
- GitHub pglogical (repository)
- Limitazioni di pglogical (file README del repository) GitHub
- Migrazione di PostgreSQL da locale o Amazon ad EC2 Amazon RDS utilizzando la replica logica (blog AWS Database)

Esegui la migrazione di un database PostgreSQL locale su Aurora PostgreSQL

Baji Shaik e Jitender Kumar, Amazon Web Services

Riepilogo

Amazon Aurora PostgreSQL Compatible Edition combina le prestazioni e la disponibilità dei database commerciali di fascia alta con la semplicità e la convenienza dei database open source. Aurora offre questi vantaggi scalando lo storage su tre zone di disponibilità nella stessa regione AWS e supporta fino a 15 istanze di replica in lettura per scalare i carichi di lavoro di lettura e fornire un'elevata disponibilità all'interno di una singola regione. Utilizzando un database globale Aurora, puoi replicare i database PostgreSQL in un massimo di cinque regioni per l'accesso in lettura remota e il disaster recovery in caso di errore di una regione. Questo modello descrive i passaggi per la migrazione di un database di origine PostgreSQL locale a un database Aurora compatibile con PostgreSQL. Il modello include due opzioni di migrazione: utilizzando AWS Data Migration Service (AWS DMS) o utilizzando strumenti PostgreSQL nativi (come pg_dump, pg_restore e psql) o strumenti di terze parti.

I passaggi descritti in questo modello si applicano anche ai database PostgreSQL di destinazione su istanze Amazon Relational Database Service (Amazon RDS) e Amazon Elastic Compute Cloud (Amazon). EC2

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database sorgente PostgreSQL in un data center locale
- Un'istanza DB Aurora compatibile con PostgreSQL o un'istanza DB Amazon RDS for PostgreSQL

Limitazioni

- I limiti di dimensione del database sono 64 TB per Amazon RDS for PostgreSQL e 128 TB per Aurora PostgreSQL compatibile.
- Se utilizzi l'opzione di migrazione AWS DMS, consulta le limitazioni di AWS DMS sull'utilizzo di un database PostgreSQL come sorgente.

Versioni del prodotto

- Per il supporto delle versioni principali e secondarie di PostgreSQL in Amazon RDS, consulta gli aggiornamenti di Amazon RDS for PostgreSQL nella documentazione di Amazon RDS.
- Per il supporto di PostgreSQL in Aurora, consulta gli aggiornamenti di Amazon Aurora PostgreSQL nella documentazione di Aurora.
- Se utilizzi l'opzione di migrazione AWS DMS, consulta le versioni PostgreSQL supportate nella documentazione di AWS DMS.

Architecture

Stack tecnologico di origine

- Database PostgreSQL locale

Stack tecnologico Target

- Istanza DB Aurora compatibile con PostgreSQL

Architettura di origine

Architettura di destinazione

Architettura di migrazione dei dati

Utilizzo di AWS DMS

Utilizzo di strumenti PostgreSQL nativi

Tools (Strumenti)

- AWS Database Migration Service (AWS DMS) ti aiuta a migrare gli archivi di dati nel cloud AWS o tra combinazioni di configurazioni cloud e locali. Questo servizio supporta diverse fonti e database di destinazione. Per informazioni su come convalidare le versioni e le edizioni del database PostgreSQL di origine e destinazione supportate per l'uso con AWS DMS, consulta Usare un database PostgreSQL come sorgente AWS DMS. Ti consigliamo di utilizzare la versione più recente di AWS DMS per il supporto più completo della versione e delle funzionalità.

- Gli strumenti nativi di PostgreSQL includono pg_dump, pg_restore e psql.

Epiche

Analizza la migrazione

Operazione	Description	Competenze richieste
Convalida le versioni del database di origine e di destinazione.	Se utilizzi AWS DMS, assicurati di utilizzare una versione supportata di PostgreSQL.	DBA
Identifica il tipo di storage e i requisiti di capacità.	1. Calcola lo storage allocato per l'istanza del database di origine. 2. Raccogli le metriche di crescita storiche per l'istanza del database di origine. 3. Anticipa le previsioni di crescita future per l'istanza di database di destinazione. 4. Alloca lo storage calcoland o il numero totale di IOPS di lettura e scrittura sul database di origine. Un volume General Purpose SSD (gp2) fornisce 3 IOPS per ogni 1 GB di storage.	DBA, amministratore di sistema
Scegli il tipo di istanza, la capacità, le funzionalità di archiviazione e le funzionalità di rete corretti.	Determina i requisiti di elaborazione dell'istanza di database di destinazi one. Esamina i problemi di prestazioni noti che potrebber o richiedere ulteriore attenzion	DBA, amministratore di sistema

Operazione	Description	Competenze richieste
	e. Considerate i seguenti fattori per determinare il tipo di istanza appropriato: • Utilizzo della CPU dell'istanza del database di origine • IOPS (operazioni di lettura e scrittura) per l'istanza del database di origine • Impronta di memoria sull'istanza del database di origine Per ulteriori informazioni, consulta le classi di istanze di Aurora DB nella documentazione di Aurora.	
Identifica i requisiti di sicurezza dell'accesso alla rete per i database di origine e di destinazione.	Determinate i gruppi di sicurezza appropriati che consentano all'applicazione di comunicare con il database.	DBA, amministratore di sistema
Identifica la strategia di migrazione delle applicazioni.	• Determina la strategia di transizione alla migrazione e in base alla complessità della tua applicazione. • Determina il Recovery Time Objective (RTO) e il Recovery Point Objective (RPO) per l'applicazione e pianifica il cutover di conseguenza.	DBA, proprietario dell'app, amministratore di sistema

Configura l'infrastruttura

Operazione	Description	Competenze richieste
Crea un VPC.	Crea un nuovo cloud privato virtuale (VPC) per l'istanza del database di destinazione.	Amministratore di sistema
Crea gruppi di sicurezza.	Crea un gruppo di sicurezza all'interno del VPC (come determinato nell'epic precedente) per consentire le connessioni in entrata all'istanza del database.	Amministratore di sistema
Configura e avvia il cluster Aurora DB.	Crea l'istanza del database di destinazione con il nuovo VPC e il nuovo gruppo di sicurezza e avvia l'istanza.	Amministratore di sistema

Migrazione dei dati – opzione 1 (utilizzando AWS DMS)

Operazione	Description	Competenze richieste
Completa i passaggi precedenti alla migrazione.	1. Pulisci i dati nel database di origine. 2. Crea un'istanza di replica. 3. Crea endpoint di origine e destinazione. 4. Identifica il numero di tabelle e oggetti disponibili da migrare.	DBA
Completa i passaggi di migrazione.	1. Elimina i vincoli e i trigger di chiave esterna nel database di destinazione.	DBA

Operazione	Description	Competenze richieste
	2. Elimina gli indici secondari sul database di destinazione. 3. Utilizza un'attività a caricamento completo per migrare i dati dal database di origine a quello di destinazione. 4. Abilita le chiavi esterne. 5. Se utilizzi la migrazione flash-cut e l'applicazione richiede tempi di inattività minimi, abilita Change Data Capture (CDC) per replicare le modifiche in corso 6. Abilita i trigger. 7. Sequenze di aggiornamento. 8. Convalida i dati di origine e di destinazione.	
Convalida i dati.	Per garantire che i dati siano stati migrati con precisione dall'origine alla destinazione, segui i passaggi di convalida dei dati nella documentazione di AWS DMS.	DBA

Migrazione dei dati – opzione 2 (usando pg_dump e pg_restore)

Operazione	Description	Competenze richieste
Preparare il database di origine.	1. Crea una directory per archiviare il backup di pg_dump se non esiste già. 2. Crea un utente di migrazione con le autorizzazioni per eseguire pg_dump sugli oggetti del database. 3. Connect all' EC2 istanza ed esegui pg_dump backup. Per ulteriori informazioni, consulta la documentazione di pg_dump e la procedura dettagliata nella documentazione di AWS DMS.	DBA
Preparare il database di destinazione.	1. Crea un utente di migrazione con le autorizzazioni per utilizzare pg_restore sugli oggetti del database. 2. Importa il dump del database utilizzando pg_restore. Per ulteriori informazioni, consulta la documentazione di pg_restore e la procedura dettagliata nella documentazione di AWS DMS.	DBA
Convalida i dati.	1. Confronta il numero di oggetti del database tra i	DBA

Operazione	Description	Competenze richieste
	database di origine e quelli di destinazione. 2. Risolvete eventuali discrepanze rilevate tra i conteggi degli oggetti.	

Migrare l'applicazione

Operazione	Description	Competenze richieste
Segui la strategia di migrazione delle applicazioni.	Implementa la strategia di migrazione delle applicazioni che hai creato nella prima epic.	DBA, proprietario dell'app, amministratore di sistema

Passa al database di destinazione

Operazione	Description	Competenze richieste
Trasferisci i client applicativi alla nuova infrastruttura.	1. Interrompi tutti i servizi applicativi e le connessioni client che puntano al database PostgreSQL locale. 2. Esegui le attività di AWS DMS. 3. Se necessario, imposta un'attività di rollback (CDC inverso da Aurora PostgreSQL compatibile con il database PostgreSQL locale). 4. Convalida i dati.	DBA, proprietario dell'app, amministratore di sistema

Operazione	Description	Competenze richieste
	5. Avvia i servizi applicativi sulla nuova destinazione configurando Amazon Route 53 sulla nuova istanza DB Aurora compatibile con PostgreSQL. 6. Aggiungi il monitoraggio di Amazon CloudWatch e Performance Insights sulla tua nuova istanza DB Aurora compatibile con PostgreSQL.	

Operazione	Description	Competenze richieste
Se è necessario ripristinare la migrazione.	1. Arresta tutti i servizi applicativi che puntano al database Aurora compatibile con PostgreSQL. 2. Ripristina le modifiche al database PostgreSQL locale di origine utilizzando il task AWS DMS creato nella storia precedente. 3. Interrompi l'esecuzione delle attività AWS DMS dal database PostgreSQL locale al database Aurora compatibile con PostgreSQL. 4. Configura l'applicazione in modo che punti al database PostgreSQL locale di origine. 5. Verifica che tutta la distribuzione del rollback sia completa.	DBA, proprietario dell'app

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse.	Chiudi le risorse AWS temporanee.	DBA, amministratore di sistema
Convalida i documenti.	Rivedi e convalida i documenti del progetto.	DBA, proprietario dell'app, amministratore di sistema

Operazione	Description	Competenze richieste
Raccogli le metriche.	Raccogli le metriche in tempo utile per la migrazione, la percentuale di risparmio sui costi manuali rispetto a quelli relativi agli strumenti e così via.	DBA, proprietario dell'app, amministratore di sistema
Chiudi il progetto.	Chiudi il progetto e fornisci eventuali feedback.	DBA, proprietario dell'app, amministratore di sistema

Risorse correlate

Riferimenti

- Servizio di migrazione dati AWS
- VPCs e Amazon Aurora
- Prezzi di Amazon Aurora
- Utilizzo di un database PostgreSQL come sorgente AWS DMS
- Come creare un'istanza di replica AWS DMS
- Come creare endpoint di origine e destinazione utilizzando AWS DMS

Altre risorse

- Guida introduttiva ad AWS DMS
- Procedure dettagliate sulla migrazione step-by-step dei dati
- Risorse Amazon Aurora

Esegui la migrazione di un database Microsoft SQL Server locale a Microsoft SQL Server su Amazon EC2 con Linux

Tirumala Dasari, Amazon Web Services

Riepilogo

Questo modello descrive come migrare da un database Microsoft SQL Server locale in esecuzione su Microsoft Windows a Microsoft SQL Server su un'istanza Amazon Elastic Compute Cloud (Amazon EC2) Linux utilizzando utilità di backup e ripristino.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- AMI Amazon EC2 Linux (Amazon Machine Image) con Microsoft SQL Server
- AWS Direct Connect tra Windows locale e Microsoft SQL Server sull'istanza Linux EC2

Architecture

Stack tecnologico di origine

- Database Microsoft SQL Server locale

Stack tecnologico Target

- EC2 Istanza Linux con un database Microsoft SQL Server

Architettura di migrazione del database

Tools (Strumenti)

- WinSCP - Questo strumento consente agli utenti Windows di condividere facilmente file con utenti Linux.
- Sqlcmd - Questa utilità da riga di comando consente di inviare istruzioni o batch T-SQL a istanze locali e remote di SQL Server. L'utilità è estremamente utile per attività ripetitive di database come l'elaborazione in batch o il test delle unità.

Epiche

Prepara l'istanza EC2 Linux con SQL Server

Operazione	Description	Competenze richieste
Seleziona un'AMI che fornisca il sistema operativo Linux e includa Microsoft SQL Server.		Amministratore di sistema
Configura l'AMI per creare un' EC2 istanza.		Amministratore di sistema
Crea regole in entrata e in uscita per i gruppi di sicurezza .		Amministratore di sistema
Configurare l' EC2 istanza Linux per un database Microsoft SQL Server.		DBA
Crea utenti e fornisci le autorizzazioni come nel database di origine.		Proprietario dell'app, DBA
Installa gli strumenti di SQL Server e l'utilità sqlcmd sull'istanza Linux. EC2		DBA

Esegui il backup del database e sposta il file di backup sull'istanza Linux EC2

Operazione	Description	Competenze richieste
Eseguire il backup del database SQL Server locale.		DBA
Installa WinSCP su Microsoft SQL Server.		DBA

Operazione	Description	Competenze richieste
Sposta il file di backup nell' EC2 istanza Linux che esegue Microsoft SQL Server.		DBA

Ripristina il database sull' EC2 istanza Linux che esegue SQL Server

Operazione	Description	Competenze richieste
Ripristina il database dal file di backup del database utilizzan do l'utilità sqlcmd.		DBA
Convalida oggetti e dati del database.		Sviluppatore, tecnico di test

Passa da Windows SQL Server a Windows SQL Server su EC2 istanza Linux

Operazione	Description	Competenze richieste
Convalida oggetti e dati del database.		Sviluppatore, tecnico di test
Passa dal database Microsoft SQL Server locale all' EC2 istanza Linux che esegue Microsoft SQL Server.		DBA

Risorse correlate

- Come configurare SQL Server 2017 su Amazon Linux e Ubuntu AMIs
- Installazione di strumenti SQL su un'istanza Linux
- Backup e ripristino da un database Microsoft SQL Server locale a Microsoft SQL Server su un'istanza Linux EC2

Esegui la migrazione di un database Microsoft SQL Server locale su Amazon RDS for SQL Server utilizzando server collegati

Kevin Yung, Viqash Adwani e Vishal Singh, Amazon Web Services

Riepilogo

I server collegati consentono a Microsoft SQL Server di eseguire istruzioni SQL su altre istanze di server di database. Questo modello descrive come migrare il database locale di Microsoft SQL Server su Amazon Relational Database Service (Amazon RDS) per Microsoft SQL Server per ottenere costi inferiori e maggiore disponibilità. Attualmente, Amazon RDS per Microsoft SQL Server non supporta connessioni al di fuori di una rete Amazon Virtual Private Cloud (Amazon VPC).

Puoi utilizzare questo modello per raggiungere i seguenti obiettivi:

- Per migrare Microsoft SQL Server ad Amazon RDS per Microsoft SQL Server senza interrompere le funzionalità dei server collegati.
- Per assegnare priorità e migrare Microsoft SQL Server collegato in diverse fasi.

Prerequisiti e limitazioni

Prerequisiti

- Verifica se Microsoft SQL Server su Amazon RDS supporta le funzionalità richieste.
- Assicurati di poter utilizzare Amazon RDS for Microsoft SQL Server con regole di confronto predefinite o regole di confronto impostate su livelli di database.

Architecture

Stack tecnologico di origine

- Database locali (Microsoft SQL Server)

Stack tecnologico Target

- Amazon RDS per SQL Server

Architettura dello stato di origine

Architettura dello stato di destinazione

Nello stato di destinazione, esegui la migrazione da Microsoft SQL Server ad Amazon RDS per Microsoft SQL Server utilizzando server collegati. Questa architettura utilizza un Network Load Balancer per inoltrare il traffico da Amazon RDS per Microsoft SQL Server ai server locali che eseguono Microsoft SQL Server. Il diagramma seguente mostra la funzionalità reverse proxy per Network Load Balancer.

Tools (Strumenti)

- AWS CloudFormation
- Network Load Balancer
- Amazon RDS per SQL Server in più zone di disponibilità (multipleAZs)
- AWS Database Migration Service (AWS DMS)

Epiche

Crea un VPC per una landing zone

Operazione	Description	Competenze richieste
Crea l'allocazione CIDR.		AWS SysAdmin
Crea un cloud privato virtuale (Virtual Private Cloud (VPC))		AWS SysAdmin
Crea le sottoreti VPC.		AWS SysAdmin

Operazione	Description	Competenze richieste
Crea gli elenchi di controllo degli accessi alle sottoreti (ACLs).		AWS SysAdmin
Crea le tabelle di routing delle sottoreti.		AWS SysAdmin
Crea una connessione con AWS Direct Connect o AWS Virtual Private Network (VPN).		AWS SysAdmin

Esegui la migrazione del database su Amazon RDS

Operazione	Description	Competenze richieste
Crea un'istanza DB Amazon RDS per Microsoft SQL Server.		AWS SysAdmin
Creare un'istanza di replica di AWS DMS.		AWS SysAdmin
Crea gli endpoint del database di origine e di destinazione in AWS DMS.		AWS SysAdmin
Crea l'attività di migrazione e imposta la replica continua su ON dopo un carico completo.		AWS SysAdmin
Richiedi una modifica del firewall per consentire ad Amazon RDS for Microsoft SQL Server di accedere ai database SQL Server locali.		AWS SysAdmin

Operazione	Description	Competenze richieste
Crea un Network Load Balancer.		AWS SysAdmin
Crea un gruppo target destinato ai server di database nel tuo data center	Ti consigliamo di utilizzare i nomi host nella configurazione di destinazione per incorpora re gli eventi di failover del data center (DC).	AWS SysAdmin
Esegui l'istruzione SQL per la configurazione del server collegato.	Esegui le istruzioni SQL per aggiungere un server collegato utilizzando lo strumento di gestione Microsoft SQL sull'istanza DB di Amazon RDS for Microsoft SQL Server. Nell'istruzione SQL, imposta @datasrc per utilizzare il nome host di Network Load Balancer. Aggiungi le credenziali di accesso al server collegato utilizzando lo strumento di gestione Microsoft SQL sull'istanza DB di Amazon RDS for Microsoft SQL Server.	AWS SysAdmin
Testa e convalida le funzioni di SQL Server.		AWS SysAdmin
Crea un cutover.		AWS SysAdmin

Risorse correlate

- Attività di gestione comuni per Microsoft SQL Server su Amazon RDS
- Regole di confronto e set di caratteri per Microsoft SQL Server

- Documentazione Network Load Balancer
- Implementazione di server collegati con Amazon RDS per Microsoft SQL Server (post sul blog)

Esegui la migrazione di un database Microsoft SQL Server locale su Amazon RDS for SQL Server utilizzando metodi di backup e ripristino nativi

Tirumala Dasari, David Queiroz e Vishal Singh, Amazon Web Services

Riepilogo

Questo modello descrive come migrare un database Microsoft SQL Server locale a un'istanza database Amazon Relational Database Service (Amazon RDS) per SQL Server DB (migrazione omogenea). Il processo di migrazione si basa su metodi di backup e ripristino nativi di SQL Server. Utilizza SQL Server Management Studio (SSMS) per creare un file di backup del database e un bucket Amazon Simple Storage Service (Amazon S3) per archiviare il file di backup prima di ripristinarlo in Amazon RDS for SQL Server.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Policy di ruolo di AWS Identity and Access Management (IAM) per accedere al bucket S3 e all'istanza DB Amazon RDS for SQL Server.

Limitazioni

- Il processo descritto in questo modello migra solo il database. Gli accessi SQL o gli utenti del database, inclusi i job di SQL Server Agent, non vengono migrati perché richiedono passaggi aggiuntivi.

Versioni del prodotto

- SQL Server 2012-2017. Per l'elenco più recente delle versioni e delle funzionalità supportate, consulta Microsoft SQL Server su Amazon RDS nella documentazione AWS.

Architecture

Stack tecnologico di origine

- Un database Microsoft SQL Server locale

Stack tecnologico Target

- Istanza database Amazon RDS per SQL Server

Architettura di migrazione dei dati

Tools (Strumenti)

- Microsoft SQL Server Management Studio (SSMS) è un ambiente integrato per la gestione dell'infrastruttura SQL Server. Fornisce un'interfaccia utente e un gruppo di strumenti con editor di script avanzati che interagiscono con SQL Server.

Epiche

Crea un'istanza DB Amazon RDS for SQL Server

Operazione	Description	Competenze richieste
Seleziona SQL Server come motore di database in Amazon RDS for SQL Server.		DBA
Scegli SQL Server Express Edition.		DBA
Specificare i dettagli del database.	Per ulteriori informazioni sulla creazione di un'istanza DB, consulta la documentazione di Amazon RDS.	DBA, proprietario dell'app

Crea un file di backup dal database SQL Server locale

Operazione	Description	Competenze richieste
Connect al database SQL Server locale tramite SSMS.		DBA

Operazione	Description	Competenze richieste
Crea un backup del database.	Per istruzioni, consulta la documentazione SSMS.	DBA, proprietario dell'app

Carica il file di backup su Amazon S3

Operazione	Description	Competenze richieste
Creare un bucket in Amazon S3.	Per ulteriori informazioni, consulta la Documentazione di Amazon S3.	DBA
Carica il file di backup nel bucket S3.	Per ulteriori informazioni, consulta la Documentazione di Amazon S3.	SysOps amministratore

Ripristina il database in Amazon RDS for SQL Server

Operazione	Description	Competenze richieste
Aggiungi il gruppo di opzioni ad Amazon RDS.	1. Aprire la console Amazon RDS all'indirizzo https://console.aws.amazon.com/rds/. 2. Nel riquadro di navigazione, scegli Gruppi di opzioni, Crea gruppo. 3. Completa le informazioni per il gruppo di opzioni, quindi scegli Crea. 4. Aggiungi l'`SQLSERVER_BACKUP_RESTORE` opzione al gruppo	SysOps amministratore

Operazione	Description	Competenze richieste
	di opzioni, quindi scegli Aggiungi opzione. Per ulteriori informazioni, consulta la documentazione di Amazon RDS.	
Ripristina il database.	1. Connect ad Amazon RDS for SQL Server tramite SSMS. 2. Richiama la `msdb.dbo.rds_restore_database` stored procedure per ripristinare il database.	DBA

Convalida il database di destinazione

Operazione	Description	Competenze richieste
Convalida oggetti e dati.	Convalida gli oggetti e i dati tra il database di origine e Amazon RDS for SQL Server. **Note** Questa attività migra solo il database. Gli accessi e i lavori non verranno migrati.	Proprietario dell'app, DBA

Tagliare

Operazione	Description	Competenze richieste
Reindirizza il traffico delle applicazioni.	Dopo la convalida, reindiriz za il traffico dell'applicazione all'istanza DB di Amazon RDS for SQL Server.	Proprietario dell'app, DBA

Risorse correlate

- Documentazione Amazon S3
- Documentazione di Amazon RDS per SQL Server
- Opzioni per il motore di database Microsoft SQL Server

Esegui la migrazione di un database Microsoft SQL Server su Aurora MySQL utilizzando AWS DMS e AWS SCT

Mark Szalkiewicz e Pavan Pusuluri, Amazon Web Services

Riepilogo

Questo modello descrive come migrare un database Microsoft SQL Server locale o su un'istanza Amazon Elastic Compute Cloud (Amazon EC2) verso Amazon Aurora MySQL. Il modello utilizza AWS Database Migration Service (AWS DMS) e AWS Schema Conversion Tool (AWS SCT) per la migrazione dei dati e la conversione dello schema.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database di origine Microsoft SQL Server in un data center locale o su un' EC2 istanza
- Driver Java Database Connectivity (JDBC) per connettori AWS SCT, installati su un computer locale o su un'istanza EC2 in cui è installato AWS SCT

Limitazioni

- Limite di dimensione del database: 64 TB

Versioni del prodotto

- Microsoft SQL Server 2008, 2008R2, 2012, 2014, 2016 e 2017 per le edizioni Enterprise, Standard, Workgroup e Developer. Le edizioni Web ed Express non sono supportate da AWS DMS. Per l'elenco più recente delle versioni supportate, consulta Using a Microsoft SQL Server Database as a Source for AWS DMS. Ti consigliamo di utilizzare la versione più recente di AWS DMS per il supporto più completo della versione e delle funzionalità. Per informazioni sulle versioni di Microsoft SQL Server supportate da AWS SCT, consulta la documentazione di AWS SCT.
- MySQL versioni 5.5, 5.6 e 5.7. Per l'elenco più recente delle versioni supportate, consulta Usare un database compatibile con MySQL come destinazione per AWS DMS.

Architecture

Stack tecnologico di origine

Una delle seguenti:

- Un database Microsoft SQL Server locale
- Un database Microsoft SQL Server su un' EC2 istanza

Stack tecnologico Target

- Aurora MySQL

Architettura di migrazione dei dati

- Da un database Microsoft SQL Server in esecuzione nel cloud AWS

- Da un database Microsoft SQL Server in esecuzione in un data center locale

Tools (Strumenti)

- AWS DMS - AWS Data Migration Service (AWS DMS) ti aiuta a migrare i dati da e verso database commerciali e open source ampiamente utilizzati, tra cui Oracle, SQL Server, MySQL e PostgreSQL. Puoi utilizzare AWS DMS per migrare i dati nel cloud AWS, tra istanze locali (attraverso la configurazione di un cloud AWS) oppure tra combinazioni di configurazioni locali e cloud.
- AWS SCT - AWS Schema Conversion Tool (AWS SCT) semplifica le migrazioni di database eterogenei convertendo automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione.

Epiche

Preparati per la migrazione

Operazione	Description	Competenze richieste
Convalida la versione e il motore del database di origine e di destinazione.		DBA
Crea un gruppo di sicurezza in uscita per i database di origine e di destinazione.		SysAdmin
Crea e configura un' EC2 istanza per AWS SCT, se necessario.		DBA
Scarica la versione più recente di AWS SCT e i driver associati.		DBA
Aggiungi e convalida gli utenti e le concessioni prerequisiti nel database di origine.		DBA
Crea un progetto AWS SCT per il carico di lavoro e connettiti al database di origine.		DBA
Genera un rapporto di valutazione e valuta la fattibilità.		DBA

Preparare il database di destinazione

Operazione	Description	Competenze richieste
Crea un'istanza database Amazon RDS di destinazione utilizzando Amazon Aurora come motore di database.		DBA
Estrai l'elenco di utenti, ruoli e concessioni dalla fonte.		DBA
Associa gli utenti esistenti del database ai nuovi utenti del database.		Proprietario dell'app
Crea utenti nel database di destinazione.		DBA
Applica i ruoli del passaggio precedente al database di destinazione.		DBA
Esamina le opzioni del database, i parametri, i file di rete e i collegamenti al database nel database di origine, quindi valuta la loro applicabilità al database di destinazione.		DBA
Applica tutte le impostazioni pertinenti all'obiettivo.		DBA

Trasferisci oggetti

Operazione	Description	Competenze richieste
Configura la connettività AWS SCT al database di destinazione.		DBA
Converti lo schema utilizzando AWS SCT.	AWS SCT converte automaticamente lo schema del database di origine e la maggior parte del codice personalizzato in un formato compatibile con il database di destinazione. Qualsiasi codice che lo strumento non è in grado di convertire automaticamente è chiaramente contrassegnato in modo che tu possa convertirlo tu stesso.	DBA
Esamina il report SQL generato e salva eventuali errori e avvisi.		DBA
Applica modifiche automatiche allo schema alla destinazione o salvale come file.sql.		DBA
Verifica che AWS SCT abbia creato gli oggetti sulla destinazione.		DBA
Riscrivi, rifiuta o riprogetta manualmente tutti gli elementi che non sono stati convertiti automaticamente.		DBA

Operazione	Description	Competenze richieste
Applica il ruolo generato e le concessioni degli utenti ed esamina le eventuali eccezioni .		DBA

Migrare i dati

Operazione	Description	Competenze richieste
Determinare il metodo di migrazione.		DBA
Crea un'istanza di replica dalla console AWS DMS.	Per informazioni dettagliate sull'uso di AWS DMS, consulta i link nella sezione «Risorse correlate».	DBA
Crea gli endpoint di origine e di destinazione.		DBA
Creare un'attività di replica.		DBA
Avvia l'attività di replica e monitora i log.		DBA

Migrare l'applicazione

Operazione	Description	Competenze richieste
Usa AWS SCT per analizzare e convertire gli elementi SQL all'interno del codice dell'appl icazione.	Quando converti lo schema del database da un motore a un altro, è anche necessario aggiornare il codice SQL nelle applicazioni, per interagire con il nuovo motore di database	Proprietario dell'app

Operazione	Description	Competenze richieste
	al posto di quello precedente. Puoi visualizzare, analizzare, modificare e salvare il codice SQL convertito. Per informazi oni dettagliate sull'uso di AWS SCT, consulta i link nella sezione «Risorse correlate».	
Crea i nuovi server delle applicazioni su AWS.		Proprietario dell'app
Esegui la migrazione del codice dell'applicazione sui nuovi server.		Proprietario dell'app
Configura il server delle applicazioni per il database e i driver di destinazione.		Proprietario dell'app
Corregge qualsiasi codice specifico del motore di database di origine dell'appl icazione.		Proprietario dell'app
Ottimizza il codice dell'appl icazione per il motore di destinazione.		Proprietario dell'app

Tagliare

Operazione	Description	Competenze richieste
Applica eventuali nuovi utenti, sovvenzioni e modifiche al codice all'obiettivo.		DBA

Operazione	Description	Competenze richieste
Blocca l'applicazione per eventuali modifiche.		Proprietario dell'app
Verifica che tutte le modifiche siano state propagate al database di destinazione.		DBA
Indirizza il nuovo server delle applicazioni verso il database di destinazione.		Proprietario dell'app
Ricontrolla tutto.		Proprietario dell'app
Trasmetti in diretta.		Proprietario dell'app

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse AWS temporanee (istanza di replica AWS DMS e EC2 istanza utilizzata per AWS SCT).		DBA, proprietario dell'app
Aggiorna il feedback sul processo AWS DMS per i team interni.		DBA, proprietario dell'app
Rivedi il processo AWS DMS e, se necessario, migliora il modello.		DBA, proprietario dell'app
Rivedi e convalida i documenti del progetto.		DBA, proprietario dell'app

Operazione	Description	Competenze richieste
Raccogli le metriche in tempo utile per la migrazione, la percentuale di risparmio sui costi manuali rispetto a quelli relativi agli strumenti e così via.		DBA, proprietario dell'app
Chiudi il progetto e fornisci eventuali feedback.		DBA, proprietario dell'app

Risorse correlate

Riferimenti

- Guida per l'utente di AWS DMS
- Guida per l'utente di AWS SCT
- Prezzi di Amazon Aurora

Tutorial e video

- Guida introduttiva ad AWS Database Migration Service
- Guida introduttiva allo Schema Conversion Tool di AWS
- Risorse Amazon RDS
- Procedure dettagliate di AWS DMS Step-by-Step

Esegui la migrazione di un database MariaDB locale su Amazon RDS for MariaDB utilizzando strumenti nativi

Shyam Sunder Rakhecha, Amazon Web Services

Riepilogo

Questo modello fornisce indicazioni per la migrazione di un database MariaDB locale ad Amazon Relational Database Service (Amazon RDS) per MariaDB utilizzando strumenti nativi. Se hai installato strumenti MySQL, puoi usare mysql e mysqldump. Se hai installato gli strumenti MariaDB, puoi usare mariadb e mariadb-dump. Gli strumenti MySQL e MariaDB hanno la stessa origine, ma ci sono piccole differenze nella versione 10.6 di MariaDB e successive.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un database di origine MariadB in un data center locale

Limitazioni

- Limite di dimensione del database: 64 TB

Versioni del prodotto

- Versioni MariaDB 10.0-10.6 (per l'elenco più recente delle versioni supportate, consulta MariaDB su Amazon RDS nella documentazione AWS)

Architecture

Stack tecnologico di origine

- Database MariaDB in un data center locale

Stack tecnologico Target

- Istanza database Amazon RDS per MariaDB

Architettura Target

Architettura di migrazione dei dati

Tools (Strumenti)

- Strumenti MySQL nativi: mysql e mysqldump
- Strumenti MariaDB nativi: mariadb e mariadb-dump

Epiche

Pianifica la migrazione

Operazione	Description	Competenze richieste
Convalida le versioni e i motori del database di origine e di destinazione.		DBA
Identifica i requisiti hardware per l'istanza del server di destinazione.		DBA, amministratore di sistema
Identifica i requisiti di archiviazione (tipo e capacità di archiviazione).		DBA, amministratore di sistema
Scegli il tipo di istanza corretto in base alla capacità, alle funzionalità di archiviazione e alle funzionalità di rete.		DBA, amministratore di sistema
Identifica i requisiti di sicurezza dell'accesso alla rete per i database di origine e di destinazione.		DBA, amministratore di sistema

Operazione	Description	Competenze richieste
Identifica la strategia di migrazione delle applicazioni.		DBA, proprietario dell'app, amministratore di sistema

Configura l'infrastruttura

Operazione	Description	Competenze richieste
Crea un cloud privato virtuale (Virtual Private Cloud (VPC))		Amministratore di sistema
Crea gruppi di sicurezza.		Amministratore di sistema
Configura e avvia un'istanza Amazon RDS DB che esegue MariaDB.		Amministratore di sistema

Migrazione dei dati

Operazione	Description	Competenze richieste
Utilizza strumenti nativi per migrare oggetti e dati del database.	Nel database di origine, usa mysqldump o mariadb-dump per creare un file di output che contenga oggetti e dati del database. Nel database di destinazione, usa mysql o mariadb per ripristinare i dati.	DBA
Convalida i dati.	Controlla i database di origine e di destinazione per confermare che la migrazione dei dati sia avvenuta correttamente.	DBA

Migrare l'applicazione

Operazione	Description	Competenze richieste
Segui la strategia di migrazione delle applicazioni.		DBA, proprietario dell'app, amministratore di sistema

Tagliare

Operazione	Description	Competenze richieste
Trasferisci i client applicativi alla nuova infrastruttura.		DBA, proprietario dell'app, amministratore di sistema

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse AWS temporanee.		Amministratore di sistema
Rivedi e convalida i documenti del progetto.		DBA, proprietario dell'app, amministratore di sistema
Raccogli le metriche relative ai tempi di migrazione, ai risparmi sui costi offerti dagli strumenti e così via.		DBA, proprietario dell'app, amministratore di sistema
Chiudi il progetto e fornisci feedback.		DBA, proprietario dell'app, amministratore di sistema

Risorse correlate

Riferimenti Amazon RDS

- Amazon RDS per MariaDB

- Amazon Virtual Private Cloud VPCs e Amazon RDS
- Implementazioni Multi-AZ di Amazon RDS
- Prezzi di Amazon RDS

Riferimenti a MySQL e MariadB

- mariadb-dump/mysqldump
- Client a riga di comando mysql

Tutorial e video

- Nozioni di base su Amazon RDS

Esegui la migrazione di un database MySQL locale su Aurora MySQL

Igor Obradovic, Amazon Web Services

Riepilogo

Questo modello spiega come migrare un database di origine MySQL locale verso Amazon Aurora MySQL Compatible Edition. Descrive due opzioni per la migrazione: utilizzando AWS Database Migration Service (AWS DMS) o utilizzando strumenti MySQL nativi come mysqldbcopy e mysqldump.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un database MySQL di origine in un data center locale

Limitazioni

- Limite di dimensione del database: 128 TB

Versioni del prodotto

- La versione 8.0 di MySQL (Aurora MySQL versione 3) è disponibile con supporto standard.
- La versione 5.7 di MySQL (Aurora MySQL versione 2) è disponibile con supporto esteso, a un costo aggiuntivo.

Per l'elenco più recente delle versioni supportate, consulta le versioni di Amazon Aurora nella AWS documentazione. Se lo stai usando AWS DMS, vedi anche Utilizzo di un database compatibile con MySQL come destinazione per le versioni di AWS DMS MySQL supportate da. AWS DMS

Architecture

Stack di tecnologia di origine

- Un database MySQL locale

Stack tecnologico Target

- Amazon Aurora edizione compatibile con MySQL

Architettura di destinazione

I dati Aurora sono archiviati in un volume cluster, che è un singolo volume virtuale che utilizza unità a stato solido ()SSDs. Un volume del cluster è composto da copie di dati distribuite su tre zone di disponibilità in una singola regione AWS. Poiché i dati vengono replicati automaticamente tra le zone di disponibilità, sono altamente durevoli con minori possibilità di perdita di dati.

Aurora divide automaticamente il volume del database in segmenti da 10 GB distribuiti su più dischi. Ogni blocco da 10 GB del volume del database viene replicato in sei modi, su tre zone di disponibilità. Il diagramma seguente illustra la relazione tra il volume del cluster, l'istanza Writer DB e le istanze DB Reader in un cluster Aurora DB e la separazione tra capacità di calcolo e archiviazione. Per ulteriori informazioni su questa architettura, consulta la documentazione e le domande frequenti di Aurora.

Architettura di migrazione dei dati

Utilizzando AWS DMS:

Il diagramma seguente illustra la migrazione di un database MySQL locale a un cluster Aurora compatibile con MySQL in, using. Cloud AWS AWS DMS

Utilizzo di strumenti MySQL nativi:

Il diagramma seguente illustra la migrazione di un database MySQL locale in un cluster Aurora compatibile con MySQL in, utilizzando strumenti MySQL nativi come mysqldbcopy e mysqldump. Cloud AWS

Tools (Strumenti)

- AWS Database Migration Service (AWS DMS) supporta diversi motori di database di origine e destinazione. Per informazioni sui database di origine e destinazione MySQL supportati AWS DMS da, vedere Migrazione di database compatibili con MySQL a. AWS Ti consigliamo di utilizzare la versione più recente di AWS DMS per il supporto più completo della versione e delle funzionalità.
- mysqldbcopy è un'utilità MySQL che copia un database MySQL su un singolo server o tra server.

- mysqldump è un'utilità MySQL che crea un file di dump da un database MySQL per scopi di backup o migrazione.

Epiche

Pianifica la migrazione

Operazione	Description	Competenze richieste
Convalida la versione e il motore.	Convalida la versione e il motore del database per i database di origine e di destinazione.	DBA
Identifica i requisiti hardware.	Identifica i requisiti hardware per l'istanza del server di destinazione.	DBA, amministratore di sistema
Identifica i requisiti di archiviazione.	Identifica i requisiti di archiviazione (tipo e capacità di archiviazione).	DBA, amministratore di sistema
Scegliere il tipo di istanza.	Scegli il tipo di istanza corretto in base ai tuoi requisiti di elaborazione, archiviazione e rete.	DBA, amministratore di sistema
Determinare i requisiti di sicurezza dell'accesso alla rete.	Identifica i requisiti di sicurezza dell'accesso alla rete per i database di origine e di destinazione.	DBA, amministratore di sistema
Determina la strategia.	Identifica la strategia di migrazione delle applicazioni.	DBA, proprietario dell'app, amministratore di sistema

Configura l'infrastruttura

Operazione	Description	Competenze richieste
Crea un cloud privato virtuale (Virtual Private Cloud (VPC))	Per istruzioni, consulta Creare un VPC nella documentazione di Amazon Virtual Private Cloud (Amazon VPC).	Amministratore di sistema
Crea gruppi di sicurezza.	Per istruzioni, consulta Creare un gruppo di sicurezza per il tuo VPC nella documentazione di Amazon VPC.	Amministratore di sistema
Configura e avvia un cluster DB Aurora compatibile con MySQL nel tuo. Account AWS	Per istruzioni, consulta Creazione di un cluster Amazon Aurora DB nella documentazione di Aurora.	Amministratore di sistema

Migrazione dei dati - opzione 1

Operazione	Description	Competenze richieste
Utilizza strumenti MySQL nativi o strumenti di terze parti per migrare oggetti e dati del database.	Per istruzioni, consulta la documentazione degli strumenti MySQL come mysqldbcopy e mysqldump.	DBA

Migrazione dei dati - opzione 2

Operazione	Description	Competenze richieste
Esegui la migrazione dei dati con AWS DMS.	Per istruzioni, vedere Utilizzo di un database compatibile con MySQL come origine e Utilizzo di un database	DBA

Operazione	Description	Competenze richieste
	compatibile con MySQL come destinazione nella documenta zione. AWS DMS	

Migrare l'applicazione

Operazione	Description	Competenze richieste
Segui la strategia.	Segui la strategia di migrazion e delle applicazioni.	DBA, proprietario dell'app, amministratore di sistema

Tagliare

Operazione	Description	Competenze richieste
Cambia client applicativo.	Passa i client dell'applicazione per connetterli al nuovo endpoint del cluster Aurora.	DBA, proprietario dell'app, amministratore di sistema

Chiudi il progetto

Operazione	Description	Competenze richieste
Chiudi le risorse.	Chiudi le AWS risorse temporanee.	DBA, amministratore di sistema
Consulta la documentazione.	Rivedi e convalida i documenti del progetto.	DBA, proprietario dell'app, amministratore di sistema
Raccogli metriche.	Raccogli le metriche relative al tempo necessario alla migrazione, alla percentuale di passaggi manuali rispetto	DBA, proprietario dell'app, amministratore di sistema

Operazione	Description	Competenze richieste
	all'utilizzo degli strumenti, ai risparmi sui costi e così via.	
Progetto di migrazione completo.	Chiudi il progetto e fornisci feedback.	Proprietario dell'app, DBA, amministratore di sistema

Risorse correlate

Riferimenti

- Migrazione dei dati al cluster Amazon Aurora MySQL DB
- AWS DMS website
- AWS DMS documentazione
- Prezzi di Amazon Aurora
- Creazione e connessione a un cluster Aurora MySQL DB
- Amazon VPC e Amazon RDS
- Documentazione Amazon Aurora

Tutorial e video

- Guida introduttiva con AWS DMS
- Guida introduttiva ad Amazon Aurora

Esegui la migrazione dei database MySQL locali su Aurora MySQL utilizzando Percona, Amazon EFS e Amazon S3 XtraBackup

Rohan Jamadagni, Udayasimha Theepireddy e sajith menon, Amazon Web Services

Riepilogo

Questo modello descrive come migrare database MySQL locali di grandi dimensioni in modo efficiente verso Amazon Aurora MySQL utilizzando Percona. XtraBackup Percona XtraBackup è un'utilità di backup open source e non bloccante per server basati su MySQL. Il modello mostra come utilizzare Amazon Elastic File System (Amazon EFS) per ridurre i tempi di caricamento del backup su Amazon Simple Storage Service (Amazon S3) e ripristinare il backup su Amazon Aurora MySQL. Il modello fornisce anche dettagli su come effettuare backup Percona incrementali per ridurre al minimo il numero di log binari da applicare al database Aurora MySQL di destinazione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Autorizzazioni per creare ruoli e policy di AWS Identity and Access Management (IAM)
- Connettività di rete tra il database MySQL locale e il cloud privato virtuale (VPC) su AWS

Limitazioni

- I server di origine devono essere sistemi basati su Linux in grado di installare un client Network File System (NFS) (nfs-utils/nfs-common).
- Il bucket S3 utilizzato per caricare i file di backup supporta solo la crittografia lato server (SSE-S3/ SSE-KMS).
- Amazon S3 limita la dimensione dei file di backup a 5 TB. Se il file di backup supera i 5 TB, puoi dividerlo in più file più piccoli.
- Il numero di file sorgente caricati nel bucket S3 non può superare il milione di file.
- Il modello supporta solo il backup XtraBackup completo e il backup incrementale di Percona. Non supporta backup parziali che utilizzano`--tables`,,, `--tables-exclude` `--tables-file`-- `databases`, `--databases-exclude` o. `--databases-file`
- Aurora non ripristina utenti, funzioni, stored procedure o informazioni sul fuso orario dal database MySQL di origine.

Versioni del prodotto

- Il database di origine deve essere MySQL versione 5.5, 5.6 o 5.7.
- Per MySQL 5.7, è necessario utilizzare Percona 2.4. XtraBackup
- Per MySQL 5.6 e 5.6, è necessario utilizzare Percona 2.3 o 2.4. XtraBackup

Architecture

Stack tecnologico di origine

- Sistema operativo basato su Linux
- Server MySQL
- Percona XtraBackup

Stack tecnologico Target

- Amazon Aurora
- Simple Storage Service (Amazon S3)
- Amazon EFS

Architettura di destinazione

Tools (Strumenti)

Servizi AWS

- Amazon Aurora è un motore di database relazionale completamente gestito che semplifica ed economica la configurazione, il funzionamento e la scalabilità delle distribuzioni MySQL. Aurora MySQL è un sostituto immediato di MySQL.
- Amazon Elastic File System (Amazon EFS) ti aiuta a creare e configurare file system condivisi nel cloud AWS.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri strumenti

- Percona XtraBackup è un'utilità open source che esegue backup in streaming, compressi e incrementali dei database MySQL senza interrompere o bloccare i database.

Epiche

Creare un file system Amazon EFS

Operazione	Description	Competenze richieste
Crea un gruppo di sicurezza da associare agli obiettivi di montaggio di Amazon EFS.	Crea un gruppo di sicurezza nel VPC configurato con un allegato VPN al database locale tramite AWS Transit Gateway. Per ulteriori informazioni sui comandi e sui passaggi descritti in questa e in altre storie, consulta i collegamenti nella sezione «Risorse correlate» alla fine di questo schema.	DevOps/database Amministratore AWS
Modifica le regole del gruppo di sicurezza.	Aggiungi una regola in entrata, utilizzando il tipo NFS, la porta 2049 e l'intervallo IP del server di database locale come origine. Per impostazione predefinita, la regola in uscita consente a tutto il traffico di uscire. In caso contrario, aggiungi una regola in uscita per aprire una connessione per la porta NFS. Aggiungi altre due regole in entrata: porta 2049 (fonte: ID del gruppo di sicurezza dello stesso gruppo di sicurezza) e porta 22 (origine: intervallo IP	DevOps/database Amministratore AWS

Operazione	Description	Competenze richieste
	da cui ti connetterai a un'istanza). EC2	
Creare un file system.	Nei target di montaggio, usa il VPC e il gruppo di sicurezza che hai creato nella storia precedente. Scegliete la modalità di throughput e le prestazioni in base I/O ai requisiti del database locale. Facoltativamente, abilita la crittografia a riposo.	DevOps/database Amministratore AWS

Installa il file system

Operazione	Description	Competenze richieste
Crea un ruolo del profilo dell'istanza IAM da associare a un' EC2 istanza.	Crea un ruolo IAM con le autorizzazioni per caricare e accedere agli oggetti in Amazon S3. Scegli il bucket S3 in cui archiviare il backup come risorsa politica.	AWS DevOps
Crea un' EC2 istanza.	Avvia un' EC2 istanza basata su Linux e collega il ruolo di profilo dell'istanza IAM creato nel passaggio precedente e il gruppo di sicurezza creato in precedenza.	AWS DevOps
Installare il client NFS.	Installa il client NFS sul server di database locale e sull' EC2 istanza. Per le istruzioni di installazione, consulta	DevOps

Operazione	Description	Competenze richieste
	la sezione «Informazioni aggiuntive».	
Montare il file system Amazon EFS.	Installa il file system Amazon EFS in locale e sull' EC2 istanza. Su ogni server, crea una directory per archiviar e il backup e monta il file system utilizzando l'endpoint di destinazione di montaggio . Per un esempio, consultat e la sezione «Informazioni aggiuntive».	DevOps

Effettuare un backup del database sorgente MySQL

Operazione	Description	Competenze richieste
Installa Percona XtraBackup.	Installa Percona XtraBacku p 2.3 o 2.4 (a seconda della versione del tuo database MySQL) sul server di database locale. Per i link di installazione, consulta la sezione «Risorse correlate».	Amministratore di database
Conta gli schemi e le tabelle nel database di origine.	Raccogli e annota il numero di schemi e oggetti nel database MySQL di origine. Utilizzerai questi conteggi per convalida re il database Aurora MySQL dopo la migrazione.	Amministratore di database
(Facoltativo) Annotate la sequenza di log binaria più	Eseguire questo passaggio se si desidera stabilire la replica	Amministratore di database

Operazione	Description	Competenze richieste
recente dal database di origine.	dei log binari tra il database di origine e Aurora MySQL per ridurre al minimo i tempi di inattività. log-bin deve essere abilitato e server_id deve essere univoco. Annotate la sequenza di log binaria corrente dal database di origine, appena prima di avviare un backup. Esegui questo passaggio appena prima del backup completo se prevedi di utilizzare solo il backup completo. Se hai intenzione di eseguire backup incrementali dopo un backup completo, esegui questo passaggio appena prima del backup incrementale finale che ripristinerai sull'istanza DB Aurora MySQL.	
Avvia un backup completo del database MySQL di origine.	Effettua un backup completo del database sorgente MySQL utilizzando Percona. XtraBackup Ad esempio, i comandi per i backup completi e incrementali, consulta la sezione «Informazioni aggiuntive».	Amministratore di database

Operazione	Description	Competenze richieste
(Facoltativo) Effettua backup incrementali utilizzando Percona. XtraBackup	I backup incrementali possono essere utilizzati per ridurre la quantità di log binari da applicare per sincronizzare il database di origine con Aurora MySQL. I database di grandi dimensioni e con un elevato numero di transazioni potrebbero generare un gran numero di log binari durante i backup. Eseguendo backup incrementali e archiviandoli su un file system Amazon EFS condiviso, puoi ridurre in modo significativo i tempi di backup e caricamento del database. Per i dettagli, consulta la sezione «Informaz ioni aggiuntive». Continua a eseguire backup increment ali finché non sei pronto per iniziare il processo di migrazione ad Aurora.	Amministratore di database

Operazione	Description	Competenze richieste
Prepara i backup.	In questa fase, i log delle transazioni vengono applicati al backup per le transazioni che erano in corso durante il backup. Continuate ad applicare i log transazionali (--apply-log-only) a ogni backup incrementale per unire i backup, ad eccezione dell'ultimo backup. Per esempi, consulta la sezione «Informazioni aggiuntive». **<efs_mount_name>**Dopo questo passaggio, il backup completo e unito sarà in ~/ / fullbackup.	Amministratore di database
Comprimi e dividi il backup finale unito.	Dopo aver preparato il backup finale unito, usa i comandi tar, zip e split per creare file compressi più piccoli del backup. Per alcuni esempi, consultate la sezione «Informazioni aggiuntive».	Amministratore di database

Ripristina il backup su un cluster Aurora MySQL DB

Operazione	Description	Competenze richieste
Carica il backup su Amazon S3.	Il file system Amazon EFS in cui sono archiviati i file di backup è montato sia sul database locale che su un' EC2 istanza, in modo che i	AWS DevOps

Operazione	Description	Competenze richieste
	file di backup siano immediata mente disponibili per l' EC2 istanza. <efs_mount_name><b ucket_name>Connettiti all' EC2 istanza utilizzando Secure Shell (SSH) e carica i file di backup compressi in un bucket S3 nuovo o esistente; ad esempio: aws s3 sync ~/ / fullbackup s3:///fullbackup. Per ulteriori dettagli, consulta i link nella sezione «Risorse correlate».	
Crea un ruolo di servizio per Aurora per accedere ad Amazon S3.	Crea un ruolo IAM con trust «rds.amazonaws.com» e una policy che consenta ad Aurora di accedere al bucket S3 in cui sono archiviati i file di backup. Le autorizzazioni richieste sono, e. ListBucket GetObject GetObjectVersion	AWS DevOps

Operazione	Description	Competenze richieste
Crea la configurazione di rete per Aurora.	Crea un gruppo di sottoreti DB del cluster con almeno due zone di disponibilità e una configurazione della tabella di routing di sottorete che consenta la connettività in uscita al database di origine. Crea un gruppo di sicurezza che consenta le connessioni in uscita al database locale e consenta agli amministratori di connettersi al cluster Aurora DB. Per ulteriori informazioni, consulta i collegamenti nella sezione «Risorse correlate».	DevOps/database Amministratore AWS
Ripristina il backup su un cluster Aurora MySQL DB.	Ripristina i dati dal backup che hai caricato su Amazon S3. Specificate la versione MySQL del vostro database di origine, fornite il nome del bucket S3 e il prefisso del percorso della cartella in cui avete caricato il file di backup (ad esempio, «fullbackup» per gli esempi nella sezione «Informazioni aggiuntive») e fornite il ruolo IAM che avete creato per autorizzare Aurora ad accedere ad Amazon S3.	DevOps/database Amministratore AWS

Operazione	Description	Competenze richieste
Convalida il database Aurora MySQL.	Convalida il conteggio dello schema e degli oggetti nel cluster Aurora DB ripristinato rispetto al conteggio ottenuto dal database di origine.	Amministratore di database
Imposta la replica binlog.	Usa la sequenza di log binaria che hai notato in precedenza, prima di eseguire l'ultimo backup ripristinato nel cluster Aurora DB. Crea un utente di replica sul database di origine e segui le istruzioni nella sezione «Informazioni aggiuntive» per fornire i privilegi appropriati, abilitare la replica su Aurora e confermare che la replica è sincronizzata.	DevOps/database Amministratore AWS

Risorse correlate

Creazione di un file system Amazon EFS

- Creazione di un gruppo di sicurezza (documentazione Amazon VPC)
- Allegati VPN Transit Gateway (documentazione Amazon VPC)
- Scalabilità del throughput VPN con AWS Transit Gateway (blog su reti e distribuzione di contenuti)
- Creazione di un file system Amazon EFS (documentazione Amazon EFS)
- Creazione di obiettivi di montaggio (documentazione Amazon EFS)
- Crittografia dei dati inattivi (documentazione Amazon EFS)

Montaggio del file system

- Ruoli IAM per Amazon EC2 (EC2 documentazione Amazon)

- Avvio di un'istanza Amazon EC2 Linux (EC2 documentazione Amazon)
- Installazione del client NFS (documentazione Amazon EFS)
- Montaggio del file system Amazon EFS su un client locale (documentazione Amazon EFS)
- Montaggio di file system EFS (documentazione Amazon EFS)

Effettuare un backup del database sorgente MySQL

- Installazione di Percona XtraBackup 2.3 (documentazione Percona) XtraBackup
- Installazione di Percona XtraBackup 2.4 (documentazione Percona) XtraBackup
- Impostazione della configurazione master di replica (documentazione MySQL)
- Migrazione dei dati da un database MySQL esterno a un cluster Aurora MySQL DB (documentazione Aurora)
- Backup XtraBackup incrementale (documentazione Percona)

Ripristino del backup su Amazon Aurora MySQL

- Creazione di un bucket (documentazione Amazon S3)
- Connessione alla tua istanza Linux tramite SSH (documentazione Amazon Ec2)
- Configurazione dell'interfaccia a riga di comando di AWS (documentazione dell'interfaccia a riga di comando di AWS)
- comando sync (riferimento ai comandi AWS CLI)
- Creazione di una policy IAM per accedere alle risorse Amazon S3 (documentazione Aurora)
- Prerequisiti del cluster DB (documentazione Aurora)
- Utilizzo dei gruppi di sottoreti DB (documentazione Aurora)
- Creazione di un gruppo di sicurezza VPC per un'istanza DB privata (documentazione Aurora)
- Ripristino di un cluster Aurora MySQL DB da un bucket S3 (documentazione Aurora)
- Configurazione della replica con MySQL o un altro cluster Aurora DB (documentazione Aurora)
- Procedura mysql.rds_set_external_master (riferimento SQL per MySQL su Amazon RDS)
- procedura mysql.rds_start_replication (riferimento SQL per MySQL su Amazon RDS)

Riferimenti aggiuntivi

- Migrazione dei dati da un database MySQL esterno a un cluster Aurora MySQL DB (documentazione Aurora)
- Download del server MySQL (sito Web Oracle)

Tutorial e video

- Migrazione dei dati MySQL a un cluster Aurora MySQL DB utilizzando Amazon S3 (AWS Knowledge Center)
- Configurazione e montaggio di Amazon EFS (video)

Informazioni aggiuntive

Installazione di un client NFS

- Se state usando Red Hat o un sistema operativo Linux simile, usate il comando:

```
$ sudo yum -y install nfs-utils
```

- Se stai usando Ubuntu o un sistema operativo Linux simile, usa il comando:

```
$ sudo apt-get -y install nfs-common
```

Per ulteriori informazioni, consulta la procedura dettagliata nella documentazione di Amazon EFS.

Montaggio del file system Amazon EFS

Usa i comandi:

```
mkdir ~/<efs_mount_name>
$ sudo mount -t nfs -o
 nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2,noresvport mount-
target-IP:/ ~/<efs_mount_name>
```

Per ulteriori informazioni, consulta la procedura dettagliata e il montaggio dei file system EFS nella documentazione di Amazon EFS.

Esecuzione di backup del database sorgente MySQL

Backup completi

Usa un comando come il seguente, che prende il backup, lo comprime e lo divide in blocchi più piccoli da 1 GB ciascuno:

```
xtrabackup --backup --user=dbuser --password=<password> --binlog-info=AUTO --stream=tar
 --target-dir=~/<efs_mount_name>/fullbackup | gzip - | split -d --bytes=1024MB - ~/
<efs_mount_name>/fullbackup/backup.tar.gz &
```

Se hai intenzione di eseguire backup incrementali successivi dopo il backup completo, non comprimere e dividere il backup. Utilizzate invece un comando simile al seguente:

```
xtrabackup --backup --user=dbuser --password=<password> --target-dir=~/
<efs_mount_name>/fullbackup/
```

Backup incrementali

Utilizza il percorso di backup completo per il `--incremental-basedir` parametro, ad esempio:

```
xtrabackup --backup --user=dbuser --password=<password> --target-dir=~/
<efs_mount_name>/incremental/backupdate --incremental-basedir=~/<efs_mount_name>/
fullbackup
```

dove basedir è il percorso del backup completo e del file xtrabackup_checkpoints.

Per ulteriori informazioni sulla creazione di backup, consulta Migrazione dei dati da un database MySQL esterno a un cluster Amazon Aurora MySQL DB nella documentazione di Aurora.

Preparazione dei backup

Per preparare un backup completo:

```
xtrabackup --prepare --apply-log-only --target-dir=~/<efs_mount_name>/fullbackup
```

Per preparare un backup incrementale:

```
xtrabackup --prepare --apply-log-only --target-dir=~/<efs_mount_name>/fullbackup --
incremental-dir=~/<efs_mount_name>/incremental/06062020
```

Per preparare il backup finale:

```
xtrabackup --prepare --target-dir=~/<efs_mount_name>/fullbackup --incremental-dir=~/
<efs_mount_name>/incremental/06072020
```

Per ulteriori informazioni, consulta Backup incrementali nella documentazione di XtraBackup Percona.

Compressione e suddivisione del backup unito

<efs_mount_name>Per comprimere il backup unito in ~/ /fullbackup:

```
tar -zcvf <backupfilename.tar.gz> ~/<efs_mount_name>/fullbackup
```

Per dividere il backup:

```
split -d -b1024M --verbose  <backupfilename.tar.gz> <backupfilename.tar.gz>
```

Configurazione della replica binlog

Per creare un utente di replica sul database di origine e fornire i privilegi appropriati:

```
CREATE USER 'repl_user'@'' IDENTIFIED BY ''; GRANT REPLICATION CLIENT, REPLICATION
 SLAVE ON *.* TO 'repl_user'@'';
```

Per abilitare la replica su Aurora collegandosi al cluster Aurora DB, abilita i log binari nel gruppo di parametri del cluster DB. Imposta `binlog_format = mixed` (è preferibile la modalità mista). Questa modifica richiede il riavvio dell'istanza per applicare l'aggiornamento.

```
CALL mysql.rds_set_external_master ('sourcedbinstanceIP', sourcedbport, 'repl_user',
 '', 'binlog_file_name', binlog_file_position, 0); CALL mysql.rds_start_replication;
```

Per confermare che la replica è sincronizzata:

```
SHOW Slave Status \G;
```

Il campo master Seconds behind mostra quanto Aurora sia indietro rispetto al database locale.

Esegui la migrazione di applicazioni Java locali su AWS utilizzando AWS App2Container

Dhananjay Karanjkar, Amazon Web Services

Riepilogo

AWS App2Container (A2C) è uno strumento a riga di comando che aiuta a trasformare le applicazioni esistenti in esecuzione su macchine virtuali in contenitori, senza bisogno di modifiche al codice. A2C rileva le applicazioni in esecuzione su un server, identifica le dipendenze e genera artefatti pertinenti per una distribuzione senza interruzioni su Amazon Elastic Container Service (Amazon ECS) e Amazon Elastic Kubernetes Service (Amazon EKS).

Questo modello fornisce i passaggi per la migrazione remota di applicazioni Java locali distribuite su un server di applicazioni su AWS Fargate o Amazon EKS utilizzando App2Container tramite la macchina di lavoro.

La macchina worker può essere utilizzata nei seguenti casi d'uso:

- L'installazione di Docker non è consentita o non è disponibile sui server delle applicazioni in cui sono in esecuzione le applicazioni Java.
- È necessario gestire la migrazione di più applicazioni distribuite su server fisici o virtuali diversi.

Questo modello utilizza AWS CodeCommit AWS CodePipeline, e AWS CodeBuild.

Prerequisiti e limitazioni

Prerequisiti

- Un server di applicazioni con un'applicazione Java in esecuzione su un server Linux
- Una macchina di lavoro con sistema operativo Linux
- Una macchina di lavoro con almeno 20 GB di spazio disponibile su disco

Limitazioni

- Non tutte le applicazioni sono supportate. Per ulteriori informazioni, consulta Applicazioni supportate per Linux.

Architecture

Stack tecnologico di origine

- Applicazioni Java in esecuzione su server Linux

Stack tecnologico Target

- AWS CodeBuild
- AWS CodeCommit
- AWS CodeDeploy
- AWS CodePipeline
- Amazon Elastic Container Registry
- AWS Fargate

Architettura Target

Tools (Strumenti)

Strumenti

- AWS App2Container — AWS App2Container (A2C) è uno strumento a riga di comando che consente di eseguire applicazioni eseguite nei data center locali o su macchine virtuali, in modo che vengano eseguite in contenitori gestiti da Amazon ECS o Amazon EKS.
- AWS CodeBuild: AWS CodeBuild è un servizio di build completamente gestito nel cloud. CodeBuild compila il codice sorgente, esegue test unitari e produce artefatti pronti per la distribuzione.
- AWS CodeCommit: AWS CodeCommit è un servizio di controllo delle versioni ospitato da Amazon Web Services che puoi utilizzare per archiviare e gestire in modo privato risorse (come documenti, codice sorgente e file binari) nel cloud.
- AWS CodePipeline: AWS CodePipeline è un servizio di distribuzione continua che puoi utilizzare per modellare, visualizzare e automatizzare i passaggi necessari per rilasciare il tuo software.
- Amazon ECS — Amazon Elastic Container Service (Amazon ECS) è un servizio di gestione dei container veloce e altamente scalabile per l'esecuzione, l'arresto e la gestione dei container su un cluster.

- Amazon ECR — Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito da AWS sicuro, scalabile e affidabile.
- Amazon EKS — Amazon Elastic Kubernetes Service (Amazon EKS) è un servizio gestito che puoi usare per eseguire Kubernetes su AWS senza dover installare, utilizzare e gestire il tuo piano di controllo o i tuoi nodi Kubernetes.
- AWS Fargate — AWS Fargate è una tecnologia che puoi usare con Amazon ECS per eseguire container senza dover gestire server o cluster di istanze Amazon Elastic Compute Cloud (Amazon). EC2 Con Fargate, non è più necessario effettuare il provisioning, configurare o dimensionare i cluster di macchine virtuali per eseguire i container.

Epiche

Imposta le credenziali

Operazione	Description	Competenze richieste
Crea un segreto per accedere al server delle applicazioni.	Per accedere al server delle applicazioni in remoto dalla macchina di lavoro, crea un segreto in AWS Secrets Manager. Per il tuo segreto, puoi utilizzare la chiave privata SSH o il certificato e la chiave privata SSH. Per ulteriori informazioni, consulta Manage secrets for AWS App2Container.	DevOps, Sviluppatore

Configurare la macchina operaia

Operazione	Description	Competenze richieste
Installa il file tar.	Esegui `sudo yum install -y tar`.	DevOps, Sviluppatore

Operazione	Description	Competenze richieste
Installare l'interfaccia a riga di comando di AWS.	Per installare Amazon Command Line Interface (AWS CLI), esegui. `curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"` Decomprimere `awscliv2.zip .` Esegui `sudo ./aws/install .`	DevOps, Sviluppatore

Installa App2Container.	Esegui i comandi seguenti:
	
	`curl -o AWSApp2Container-installer-linux.tar.gz https://app2container-release-us-east-1.s3.us-east-1.amazonaws.com/latest/linux/AWSApp2Container-installer-linux.tar.gz`
	
	`sudo tar xvf AWSApp2Container-installer-linux.tar.gz`
	
	`sudo ./install.sh`
		DevOps, Sviluppatore

Operazione	Description	Competenze richieste
Configura i profili.	Per configurare il profilo predefinito di AWS, eseguisudo aws configure . Per configurare il profilo predefinito denominato AWS, eseguisudo aws configure --profile <profile name>.	DevOps, Sviluppatore
Installazione di Docker.	Esegui i comandi seguenti. sudo yum install -y docker sudo systemctl enable docker & sudo systemctl restart docker	

Operazione	Description	Competenze richieste
Inizializza App2Container.	Per inizializzare App2Container, sono necessarie le seguenti informazioni: • `workspace` : per archiviare gli artefatti della containerizzazione delle applicazioni. Si consiglia di fornire un percorso di directory con almeno 20 GB di spazio libero su disco. • `awsProfile` : profilo AWS configurato sul server. Ciò è necessario per caricare artefatti su Amazon S3, eseguire `containerize` il comando e generare artefatti AWS per la distribuzione su Amazon ECS o Amazon EKS. • `s3Bucket`: per estrarre e archiviare artefatti AWS. • `metricsReportPermission` : Per raccogliere e archiviare le metriche riportate. • `dockerContentTrust` : Per firmare l'immagine Docker. Esegui `sudo app2container init` .	DevOps, Sviluppatore

Configurare la macchina di lavoro

Operazione	Description	Competenze richieste
Configura la macchina di lavoro per connettersi in remoto ed eseguire i comandi App2Container sul server delle applicazioni.	Per configurare la macchina di lavoro, sono necessarie le seguenti informazioni: • `Server FQDN`: il nome di dominio completo del server delle applicazioni. • `Server IP address`: l'indirizzo IP del server delle applicazioni. L'FQDN o l'indirizzo IP sono sufficienti. • `SecretARN` : L'Amazon Resource Name (ARN) del segreto utilizzato per connettersi al server delle applicazioni e archiviato in Secrets Manager. • `AuthMethod` : Il metodo di `cert` autenticazione `key` o. Esegui `sudo app2container remote configure` .	DevOps, Sviluppatore

Scopri, analizza ed estrai le applicazioni sulla macchina operatrice

Operazione	Description	Competenze richieste
Scopri le applicazioni Java locali.	Per scoprire in remoto tutte le applicazioni in esecuzione	Sviluppatore, DevOps

Operazione	Description	Competenze richieste
	sul server delle applicazioni, esegui il comando seguente. `sudo app2container remote inventory --target <FQDN/IP of App server>` Questo comando genera un elenco di applicazioni distribui te in. `inventory.json`	
Analizza le applicazioni scoperte.	Per analizzare in remoto ogni applicazione utilizzando l'application-id ottenuto nella fase di inventario, esegui il comando seguente. `sudo app2container remote analyze --application-id <java-app-id> --target <FQDN/IP of App Server>` Questo genera il `analysis.json` file nella posizione dell'area di lavoro. Dopo aver generato questo file, puoi modificare i parametri di containerizzazione in base alle tue esigenze.	Sviluppatore, DevOps

Operazione	Description	Competenze richieste
Estrarre le applicazioni analizzate.	Per generare un archivio applicativo per l'applicazione analizzata, esegui in remoto il comando seguente, che genererà il pacchetto tar nella posizione dell'area di lavoro. `sudo app2container remote extract --application-id <application id> --target <FQDN/IP of App Server>` Gli artefatti estratti possono essere generati sulla macchina di lavoro locale.	Sviluppatore, DevOps

Containerizza gli artefatti estratti sulla macchina operaia

Operazione	Description	Competenze richieste
Containerizza gli artefatti estratti.	Containerizzate gli artefatti estratti nel passaggio precedente eseguendo il comando seguente. `sudo app2container containerize --input-archive <tar bundle location on worker machine>`	Sviluppatore, DevOps
Finalizza l'obiettivo.	Per finalizzare l'obiettivo, apri`deployment.json` , che	Sviluppatore, DevOps

Operazione	Description	Competenze richieste
	viene creato all'esecuzione del `containerize` comando. Per specificare AWS Fargate come destinazione, imposta `sucreateEcsArtifacts` . `true` Per impostare Amazon EKS come obiettivo, imposta su `createEksArtifacts` true.	

Genera ed esegui il provisioning di artefatti AWS

Operazione	Description	Competenze richieste
Genera artefatti di distribuzione AWS sulla macchina di lavoro.	Per generare artefatti di distribuzione, esegui il comando seguente. `sudo app2container generate app-deployment --application-id <application id>` Questo genera il CloudFormation modello `ecs-master.yml` AWS nell'area di lavoro.	DevOps
Fornisci gli artefatti.	Per fornire ulteriormente gli artefatti generati, distribuisci il CloudFormation modello AWS eseguendo il comando seguente.	DevOps

Operazione	Description	Competenze richieste
	`aws cloudformation deploy --template-file <path to ecs-master.yml> --capabilities CAPABILITY_NAMED_IAM --stack-name <application id>-ECS`	
Genera la pipeline.	Modify`pipeline.json` , creato nella storia precedente, in base alle tue esigenze. Quindi esegui il `generate pipeline` comando per generare gli artefatti di distribuzione della pipeline.	DevOps

Risorse correlate

- Che cos'è App2Container?
- Post sul blog di AWS App2Container
- Nozioni di base sulla configurazione dell'interfaccia a riga di comando di AWS
- Nozioni di base su Docker per Amazon ECS
- Comandi Docker

Migrazione di file system condivisi in una migrazione di AWS grandi dimensioni

Amit Rudraraju, Sam Apa, Bheemeswararao Balla, Wally Lu e Sanjeev Prakasam, Amazon Web Services

Riepilogo

La migrazione di 300 o più server è considerata una migrazione di grandi dimensioni. Lo scopo di una migrazione su larga scala è migrare i carichi di lavoro dai data center locali esistenti a Cloud AWS, e questi progetti si concentrano in genere sui carichi di lavoro di applicazioni e database. Tuttavia, i file system condivisi richiedono un'attenzione mirata e un piano di migrazione separato. Questo modello descrive il processo di migrazione per i file system condivisi e fornisce le migliori pratiche per migrarli con successo nell'ambito di un progetto di migrazione di grandi dimensioni.

Un file system condiviso (SFS), noto anche come file system di rete o cluster, è una condivisione di file montata su più server. L'accesso ai file system condivisi avviene tramite protocolli come Network File System (NFS), Common Internet File System (CIFS) o Server Message Block (SMB).

Questi sistemi non vengono migrati con strumenti di migrazione standard, ad esempio AWS Application Migration Service perché non sono dedicati all'host oggetto della migrazione né sono rappresentati come dispositivi a blocchi. Sebbene la maggior parte delle dipendenze degli host venga migrata in modo trasparente, il coordinamento e la gestione dei file system dipendenti devono essere gestiti separatamente.

La migrazione dei file system condivisi avviene nelle seguenti fasi: individuazione, pianificazione, preparazione, suddivisione e convalida. Utilizzando questo modello e le cartelle di lavoro allegate, migri il tuo file system condiviso verso un servizio di AWS storage, come Amazon Elastic File System (Amazon EFS), Amazon FSx for NetApp ONTAP o Amazon FSx for Windows File Server. Per trasferire il file system, puoi utilizzare AWS DataSync uno strumento di terze parti, ad esempio. NetApp SnapMirror

> **Note**
>
> Questo modello fa parte di una serie di AWS Prescriptive Guidance sulle migrazioni di grandi dimensioni verso. Cloud AWS Questo modello include le migliori pratiche e istruzioni da SFSs incorporare nei piani wave per i server. Se stai migrando uno o più file system condivisi al di fuori di un progetto di migrazione di grandi dimensioni, consulta le istruzioni per il trasferimento dei dati nella AWS documentazione per Amazon EFS, Amazon FSx for Windows File Server e Amazon FSx for NetApp ONTAP.

Prerequisiti e limitazioni

Prerequisiti

I prerequisiti possono variare in base ai file system condivisi di origine e destinazione e al caso d'uso. I più comuni sono i seguenti:

- Un attivo Account AWS.
- Avete completato l'esplorazione del portafoglio di applicazioni per il vostro grande progetto di migrazione e avete iniziato a sviluppare piani d'ondata. Per ulteriori informazioni, consulta Portfolio playbook per migrazioni di AWS grandi dimensioni.
- Cloud privati virtuali (VPCs) e gruppi di sicurezza che consentono il traffico in ingresso e in uscita tra il data center locale e l'ambiente. AWS Per ulteriori informazioni, consulta Opzioni di connettività da rete ad Amazon VPC e requisiti di rete.AWS DataSync
- Autorizzazioni per creare AWS CloudFormation stack o autorizzazioni per creare risorse Amazon EFS o Amazon. FSx Per ulteriori informazioni, consulta la CloudFormation documentazione, la documentazione di Amazon EFS o la FSx documentazione di Amazon.
- Se utilizzi AWS DataSync per eseguire la migrazione, hai bisogno delle seguenti autorizzazioni:
  - Autorizzazioni per AWS DataSync inviare log a un gruppo di log di Amazon CloudWatch Logs. Per ulteriori informazioni, consulta Consentire il caricamento dei log DataSync in gruppi di log. CloudWatch
  - Autorizzazioni per accedere al gruppo CloudWatch Logs log. Per ulteriori informazioni, vedere Panoramica della gestione delle autorizzazioni di accesso alle risorse Logs. CloudWatch
  - Autorizzazioni per creare agenti e attività in. DataSync Per ulteriori informazioni, consulta Autorizzazioni IAM richieste per l'utilizzo. AWS DataSync

Limitazioni

- Questo modello è progettato per migrare SFSs come parte di un progetto di migrazione di grandi dimensioni. Include le migliori pratiche e istruzioni da SFSs incorporare nei piani Wave per la migrazione delle applicazioni. Se stai migrando uno o più file system condivisi al di fuori di un progetto di migrazione di grandi dimensioni, consulta le istruzioni per il trasferimento dei dati nella AWS documentazione per Amazon EFS, Amazon FSx for Windows File Server e Amazon FSx for NetApp ONTAP.
- Questo modello si basa su architetture, servizi e modelli di migrazione di uso comune. Tuttavia, i progetti e le strategie di migrazione di grandi dimensioni possono variare tra le organizzazioni.

Potrebbe essere necessario personalizzare questa soluzione o le cartelle di lavoro fornite in base alle proprie esigenze.

Architecture

Stack tecnologico di origine

Uno o più dei seguenti:

- File server Linux (NFS)
- File server Windows (SMB)
- NetApp array di storage
- array di storage Dell EMC Isilon

Stack tecnologico Target

Uno o più dei seguenti:

- Amazon Elastic File System
- Amazon FSx per NetApp ONTAP
- File server Amazon FSx per Windows

Architettura Target

Il diagramma mostra il seguente processo:

1. Si stabilisce una connessione tra il data center locale e il Cloud AWS utilizzando un sistema Servizio AWS come AWS Direct Connect o. AWS Site-to-Site VPN
2. L' DataSync agente viene installato nel data center locale.
3. In base al vostro piano d'azione, dovete DataSync replicare i dati dal file system condiviso di origine alla condivisione di file di destinazione AWS .

Fasi di migrazione

L'immagine seguente mostra le fasi e i passaggi di alto livello per la migrazione di un SFS in un progetto di migrazione di grandi dimensioni.

La sezione Epics di questo modello contiene istruzioni dettagliate su come completare la migrazione e utilizzare le cartelle di lavoro allegate. Di seguito è riportata una panoramica di alto livello delle fasi di questo approccio graduale.

Fase	Fasi
Scopri	1. Utilizzando uno strumento di rilevamento, raccogli dati sul file system condiviso, inclusi server, punti di montaggio e indirizzi IP. 2. Utilizzando un database di gestione della configurazione (CMDB) o lo strumento di migrazione, si raccolgono dettagli sul server, tra cui informazioni sull'ondata di migrazione, sull'ambiente, sul proprietario dell'applicazione, sul nome del servizio di gestione dei servizi IT (ITSM), sull'unità organizzativa e sull'ID dell'applicazione.
Pianificazione	3. Utilizzando le informazioni raccolte su SFSs e sui server, create il piano d'onda SFS. 4. Utilizzando le informazioni nel foglio di lavoro di compilazione, per ogni SFS, scegliete una destinazione Servizio AWS e uno strumento di migrazione.
Preparazione	5. Configura l'infrastruttura di destinazione in Amazon EFS, Amazon FSx for NetApp ONTAP o Amazon FSx for Windows File Server. 6. Configura il servizio di trasferimento dati, ad esempio DataSync, e quindi avvia la sincroniz zazione iniziale dei dati. Una volta completat a la sincronizzazione iniziale, puoi configurare sincronizzazioni ricorrenti da eseguire secondo una pianificazione,

	7. Aggiorna il piano d'onda SFS con informazioni sulla condivisione del file di destinazione, come l'indirizzo IP o il percorso.
Tagliare	8. Blocca le applicazioni che accedono attivamente all'SFS di origine. 9. Nel servizio di trasferimento dati, eseguite una sincronizzazione finale dei dati. 10. Una volta completata la sincronizzazione, verifica che sia avvenuta correttamente esaminando i dati di registro in CloudWatch Logs.
Convalida	11. Sui server, modificate il punto di montaggio sul nuovo percorso SFS. 12. Riavviare e convalidare le applicazioni.

Tools (Strumenti)

Servizi AWS

- Amazon CloudWatch Logs ti aiuta a centralizzare i log di tutti i tuoi sistemi e applicazioni, Servizi AWS così puoi monitorarli e archiviarli in modo sicuro.
- AWS DataSyncè un servizio di trasferimento e scoperta di dati online che ti aiuta a spostare file o dati di oggetti da, verso e tra i servizi di storage. AWS
- Amazon Elastic File System (Amazon EFS) ti aiuta a creare e configurare file system condivisi nel Cloud AWS.
- Amazon FSx fornisce file system che supportano i protocolli di connettività standard del settore e offrono disponibilità e replica elevate in tutto il mondo. Regioni AWS

Altri strumenti

- SnapMirrorè uno strumento di replica NetApp dei dati che replica i dati da volumi o qtree di origine specificati rispettivamente su volumi o qtree di destinazione. Puoi utilizzare questo strumento per migrare un file system di NetApp origine su Amazon FSx for NetApp ONTAP.

- Robocopy, che è l'abbreviazione di Robust File Copy, è una directory a riga di comando e un comando per Windows. Puoi utilizzare questo strumento per migrare un file system di origine di Windows su Amazon FSx for Windows File Server.

Best practice

Approcci alla pianificazione delle onde

Quando pianificate ondate per un progetto di migrazione di grandi dimensioni, tenete conto della latenza e delle prestazioni delle applicazioni. Quando l'SFS e le applicazioni dipendenti operano in luoghi diversi, ad esempio uno nel cloud e l'altro nel data center locale, ciò può aumentare la latenza e influire sulle prestazioni delle applicazioni. Le seguenti sono le opzioni disponibili per la creazione di piani ondulatori:

1. Migra l'SFS e tutti i server dipendenti all'interno della stessa ondata: questo approccio previene problemi di prestazioni e riduce al minimo le rilavorazioni, come la riconfigurazione dei punti di montaggio più volte. È consigliato quando è richiesta una latenza molto bassa tra l'applicazione e l'SFS. Tuttavia, la pianificazione delle ondate è complessa e l'obiettivo in genere è rimuovere le variabili dai raggruppamenti di dipendenze, anziché aggiungerle. Inoltre, questo approccio non è consigliato se molti server accedono allo stesso SFS perché rende l'onda troppo grande.
2. Eseguite la migrazione dell'SFS dopo la migrazione dell'ultimo server dipendente: ad esempio, se più server accedono a un SFS e tali server sono programmati per migrare nelle ondate 4, 6 e 7, pianificate la migrazione dell'SFS nell'ondata 7.

   Questo approccio è spesso il più logico per le migrazioni di grandi dimensioni ed è consigliato per le applicazioni sensibili alla latenza. Riduce i costi associati al trasferimento dei dati. Inoltre, riduce al minimo il periodo di latenza tra SFS e le applicazioni di livello superiore (come la produzione), poiché le applicazioni di livello superiore sono in genere programmate per migrare per ultime, dopo lo sviluppo e le applicazioni di controllo qualità.

   Tuttavia, questo approccio richiede ancora scoperta, pianificazione e agilità. Potrebbe essere necessario migrare l'SFS in un'ondata precedente. Verificate che le applicazioni siano in grado di sopportare la latenza aggiuntiva per il periodo di tempo compreso tra la prima onda dipendente e l'onda contenente l'SFS. Conduci una sessione di rilevamento con i proprietari delle applicazioni e migra l'applicazione nella stessa ondata, l'applicazione più sensibile alla latenza. Se vengono rilevati problemi di prestazioni dopo la migrazione di un'applicazione dipendente, preparatevi a passare rapidamente alla migrazione SFS il più rapidamente possibile.

3. Migrate l'SFS al termine di un ampio progetto di migrazione: questo approccio è consigliato se la latenza non è un fattore, ad esempio quando i dati nell'SFS sono accessibili di rado o non sono critici per le prestazioni dell'applicazione. Questo approccio semplifica la migrazione e semplifica le attività di cutover.

È possibile combinare questi approcci in base alla sensibilità alla latenza dell'applicazione. Ad esempio, è possibile migrare sensibili alla latenza SFSs utilizzando gli approcci 1 o 2 e quindi migrare il resto utilizzando l'approccio 3. SFSs

Scelta di un servizio di file system AWS

AWS offre diversi servizi cloud per l'archiviazione di file. Ciascuno offre vantaggi e limiti diversi in termini di prestazioni, scalabilità, accessibilità, integrazione, conformità e ottimizzazione dei costi. Esistono alcune opzioni logiche predefinite. Ad esempio, se il tuo attuale file system locale utilizza Windows Server, Amazon FSx for Windows File Server è la scelta predefinita. Oppure, se il file system locale utilizza NetApp ONTAP, Amazon FSx for NetApp ONTAP è la scelta predefinita. Tuttavia, potresti scegliere un servizio mirato in base ai requisiti della tua applicazione o per ottenere altri vantaggi operativi sul cloud. Per ulteriori informazioni, consulta Scelta del servizio di archiviazione dei AWS file giusto per la propria implementazione (presentazione AWS Summit).

Scelta di uno strumento di migrazione

Amazon EFS e Amazon FSx supportano l'uso di AWS DataSync per migrare i file system condivisi verso Cloud AWS. Per ulteriori informazioni sui sistemi e servizi di storage supportati, sui vantaggi e sui casi d'uso, consulta What is AWS DataSync. Per una panoramica del processo di utilizzo DataSync del trasferimento dei file, consulta Come funzionano AWS DataSync i trasferimenti.

Sono disponibili anche diversi strumenti di terze parti, tra cui:

- Se scegli Amazon FSx for NetApp ONTAP, puoi utilizzarlo NetApp SnapMirror per migrare i file dal data center locale al cloud. SnapMirror utilizza la replica a livello di blocco, che può essere più veloce DataSync e ridurre la durata del processo di trasferimento dei dati. Per ulteriori informazioni, consulta Migrazione a FSx for ONTAP using. NetApp SnapMirror
- Se scegli Amazon FSx for Windows File Server, puoi usare Robocopy per migrare i file nel cloud. Per ulteriori informazioni, consulta Migrazione di file esistenti su Windows File Server FSx utilizzando Robocopy.

Poemi epici

Scopri

Operazione	Description	Competenze richieste
Preparate la cartella di lavoro SFS Discovery.	1. Scarica le cartelle di lavoro nella sezione Allegati di questo modello. Contiene due file, SFS-Discovery-Workbook.xlsx e SFS-Wave-Plan-Workbook.xlsx. 2. Aprire il file SFS-Disco very-Workbook in Microsoft Excel. 3. Nel foglio di lavoro Dashboard, effettuate le seguenti operazioni: • Nella colonna A, aggiorna il nome dell'ambiente. • Nella colonna B, aggiorna l'ordine degli ambienti per metterli in ordine di priorità più bassa (1) a priorità più alta. • Nelle colonne D—E, aggiorna la pianificazione delle onde. • Nelle colonne C e K, aggiorna i nomi degli account AWS. • Nella colonna L, aggiorna il VPC IDs. • Nelle colonne M—O, aggiorna la sottorete. IDs	Ingegnere addetto alla migrazione, responsabile della migrazione

Operazione	Description	Competenze richieste
	4. Rivedi il resto del modello di cartella di lavoro e aggiorna gli altri valori necessari per l'organizzazione o il caso d'uso. 5. Salva la cartella di lavoro.	

Operazione	Description	Competenze richieste
Raccogli informazioni sulla fonte SFS.	1. Utilizzando il vostro strumento di rilevamento preferito, identificate tutti i supporti SFS su tutti i dispositivi di storage, i server Linux e i server Windows applicabili. In genere, è necessario raccogliere le seguenti informazioni: • Dispositivi client • Indirizzo IP del client • Dettagli SFS • Punto di montaggio **Note** È possibile aggiungere dettagli sul punto di montaggio al runbook di migrazione per rimontare l'SFS dopo la migrazione. 2. Aprire il file SFS-Discovery-Workbook. 3. Sul foglio di lavoro Wave-Sheet, effettuate le seguenti operazioni: • Nella colonna Posizione del server (D), nella	Ingegnere addetto alla migrazione, responsabile della migrazione

Operazione	Description	Competenze richieste
	formula, conferma che il formato dell'inte rvallo CIDR per l'origine locale sia adatto al tuo intervallo. Ad esempio, se l'intervallo CIDR è`10.0.0.0/8` , inserisci `10.*.*.*` . • Nella colonna SFS location (E), nella formula, conferma che il formato dell'intervallo CIDR per il VPC di destinazione funzioni per il tuo intervallo. Ad esempio, se l'interva llo CIDR è, inserisci `176.16.0.0/16` `176.16.*.*` . 4. Nel foglio di lavoro SFS-Data, effettuate le seguenti operazioni: • Nella colonna Nome server (A), inserite il nome del server su cui è montato l'SFS. • Nella colonna Percorso SFS (B), inserite il nome dell'SFS. • Nella colonna Indirizzo IP (C), inserite l'indirizzo IP del server. • Aggiungete tutte le altre informazioni pertinent	

Operazione	Description	Competenze richieste
	i raccolte durante il rilevamento, ad esempio il punto di montaggio e la dimensione SFS. È possibile utilizzare questi dati in un secondo momento per modificare i calcoli di pianificazione delle ondate. 5. Salva la cartella di lavoro.	

Operazione	Description	Competenze richieste
Raccogli informazioni sui server.	1. Utilizzando il CMDB o i dati registrati nello strumento di migrazione, identificate tutte le seguenti informazioni sui server dotati di supporti SFS: • Server name (Nome del server) • IP address (Indirizzo IP) • Onda • Unità organizzativa (UO) • Ambiente server, ad esempio `DEVQA`, o `PROD` • Application name (Nome applicazione) • Proprietario dell'appl icazione e informazioni di contatto 2. Aprire il file SFS-Discovery-Workbook. 3. Nel foglio di lavoro Server-Data, nelle colonne A—H, inserisci le informazioni che hai raccolto sui server di origine. Tenere presente quanto segue: • Nella colonna Wave # (C), immettete il nome dell'onda (ad esempio`Wave1`), out-of-scope () o. `OOS` `Retire`	Ingegnere addetto alla migrazione, responsabile della migrazione

Operazione	Description	Competenze richieste
	• Se la colonna Contatti del proprietario dell'app (H), verifica che l'indiriz zo email sia corretto. Questo indirizzo email viene generato automatic amente in base al nome fornito nella colonna Proprietario dell'app (G). Se necessario, aggiorna manualmente il valore in modo che corrispon da all'indirizzo e-mail corretto. • Non modificare le colonne I—J, che contengono formule. 4. Salva la cartella di lavoro.	

Pianificazione

Operazione	Description	Competenze richieste
Costruisci il piano d'onda SFS.	1. Aprire il file SFS-Discovery-Workbook. 2. Verifica che tutte le informazioni raccolte nella fase di scoperta siano accurate e aggiornate. 3. Nel foglio di lavoro Wave-Sheet, filtra la colonna SFS wave (K) sul valore. 1 Questo è un elenco di tutti	Responsabile sviluppo, responsabile Cutover, ingegnere addetto alla migrazione, responsabile della migrazione

Operazione	Description	Competenze richieste
	i componenti della prima ondata SFSs . **Note** Un valore `0` in questa colonna indica che l'SFS non rientra nell'ambito della migrazione. Ciò potrebbe essere dovuto al fatto che l'SFS è già ospitato su AWS o perché i server che accedono alla condivisione non rientrano nell'ambito della migrazione. 4. Verifica di volerli migrare SFSs in questa ondata. Per ulteriori informazioni su come assegnare le ondate, consulta Approcci SFSs alla pianificazione delle ondate nella sezione Best Practice. 5. Seleziona e copia le celle contenenti i valori filtrati. Non copiate la riga di intestazione contenente i titoli delle colonne. 6. Aprite il file SFS-Wave-Plan-Workbook che avete scaricato in precedenza.	

Operazione	Description	Competenze richieste
	7. Nel foglio di lavoro Export-From-Discovery, selezionare la cella A2. 8. Incolla i dati copiati. 9. Salvate i file SFS-Discovery-Workbook e SFS-Wave-Plan-Workbook.	

Operazione	Description	Competenze richieste
Scegli lo strumento di destinazione Servizio AWS e di migrazione.	1. Nel SFS-Wave-Plan-Work book file, sul Exported-from-Discovery foglio di lavoro, seleziona e copia i valori nella colonna Old path (C). 2. Nel foglio di lavoro Build-Wave, seleziona la cella A2. 3. Incolla i dati copiati. Le colonne B—M di questo foglio di lavoro si aggiornan o automaticamente per riflettere altri dati associati a questo percorso. 4. Rimuovi tutti i valori duplicati nella colonna A. Per istruzioni, vedi Rimuovere valori duplicati (sito Web Microsoft Support). 5. Nella colonna Target pattern or service (F), esamina la destinazione consigliata Servizio AWS e aggiornala se necessari o. Per ulteriori informazi oni, consulta Scelta di un servizio di AWS file system nella sezione Best practice di questo modello. 6. Nella colonna Metodo di migrazione (G), esaminate lo strumento di migrazione consigliato e aggiornatelo	Ingegnere addetto alla migrazione, responsabile della migrazione

Operazione	Description	Competenze richieste
	se necessario. Per ulteriori informazioni, consulta Scelta di uno strumento di migrazione nella sezione Best practice di questo modello. 7. Salvate il file SFS-Disco very-Workbook. Hai finito di creare un piano d'onda per questa ondata. 8. Ripeti queste istruzioni per preparare un piano d'onda per ogni onda. Poiché i piani ondulatori sono soggetti a modifiche durante la migrazione, consigliamo di pianifica re non più di 5 ondate in anticipo.	

Preparazione

Operazione	Description	Competenze richieste
Configura il file system di destinazione.	In base ai dettagli registrati nel tuo piano d'onda, configura i file system di destinazione nel target Account AWS, nel VPC e nelle sottoreti. Per istruzioni, consulta la seguente documentazione: AWS • Amazon EFS	Ingegnere addetto alla migrazione, responsabile della migrazione, amministratore AWS

Operazione	Description	Competenze richieste
	• Amazon FSx per NetApp ONTAP • File server Amazon FSx per Windows	

Operazione	Description	Competenze richieste
Configura lo strumento di migrazione e trasferisci i dati.	1. Se lo utilizzi AWS DataSync, configura la registrazione per le DataSync attività. Per istruzioni, consulta Registrazione delle attività delle AWS DataSync attività. 2. Configura lo strumento di migrazione ed esegui un trasferimento iniziale dei dati in base alle istruzioni dello strumento selezionato: • Per Amazon EFS, consulta quanto segue: • Trasferisci file su Amazon EFS utilizzando AWS DataSync • Per Amazon FSx for NetApp ONTAP, consulta quanto segue: • Migrazione a FSx for ONTAP utilizzando NetApp SnapMirror • Migrazione a for ONTAP FSx utilizzando AWS DataSync • Per Amazon FSx for Windows File Server, consulta quanto segue: • Migrazione di file esistenti su FSx Windows File Server	Amministratore AWS, amministratore cloud, ingegnere addetto alla migrazione, responsabile della migrazione

Operazione	Description	Competenze richieste
	utilizzando AWS DataSync • Migrazione di file esistenti su FSx Windows File Server utilizzando Robocopy 3. Le modifiche all'SFS di origine potrebbero verificarsi durante o dopo il trasferimento iniziale. Imposta trasferimenti di dati ricorrenti tra i file system di origine e di destinazione per mantenere i dati sincroniz zati: • Se lo stai utilizzando DataSync, vedi Pianifica zione delle attività. AWS DataSync DataSync trasferisce solo i file nuovi o modificati nell'SFS di origine. • Se utilizzi uno strumento di terze parti, consulta la documentazione dello strumento selezionato.	

Operazione	Description	Competenze richieste
Aggiorna il piano d'ondata.	1. Aprire il file SFS-Wave-Plan-Workbook per l'onda corrente. 2. Nel foglio di lavoro Build —Wave, nella colonna Nuovo percorso IP address (N), inserisci l'indirizzo IP del file system di destinazi one. Effettuare una delle seguenti operazioni per individuare l'indirizzo IP: • FSx Per Windows File Server, sulla FSx console Amazon, scegli File system, scegli il tuo file system, quindi visualizz a la sezione Rete e sicurezza. • FSx Per ONTAP, vedi Volumi di montaggio. • Per Amazon EFS, consulta Montaggio con un indirizzo IP. 3. Nella colonna Nuovo percorso (O), inserisci il nuovo percorso di montaggio. Il percorso di montaggio è il nome DNS del file system. Effettuat e una delle seguenti operazioni per individuare il percorso di montaggio:	Ingegnere addetto alla migrazione, responsabile della migrazione

Operazione	Description	Competenze richieste
	• FSx Per Windows File Server, sulla FSx console Amazon, scegli File system, scegli il tuo file system e quindi scegli Allega. • FSx Per ONTAP, consulta la pagina dei dettagli del file system. Per istruzioni, vedere Volumi di montaggio. • Per Amazon EFS, consulta Gather Information. 4. Nel foglio di lavoro Remount-Summary, verifica che le colonne New path (C) e New path IP address (D) riflettano i valori aggiornati. 5. Verificate che l'organizzazione abbia preparato i runbook per il rimontaggio dei file system Linux e Windows dopo il cutover. Per istruzioni generali, consultate quanto segue: • Montaggio di file system Amazon EFS • Accesso FSx alle condivisioni di file Windows File Server	

Operazione	Description	Competenze richieste
	• Montaggio FSx per volumi ONTAP 6. Se in questa ondata non sono inclusi server dipendenti, registrali nel foglio di lavoro App-Team-Communication. Informate i rispettivi proprietari delle applicazioni o dei server perché potrebber o non essere inclusi nelle comunicazioni wave standard. 7. Se SFSs vengono rimossi dall'onda dopo aver completato il piano d'onda, tenetene traccia nel foglio di lavoro Descoped.	

Tagliare

Operazione	Description	Competenze richieste
Interrompi le applicazioni.	Se le applicazioni o i client eseguono attivamente operazioni di lettura e scrittura nell'SFS di origine, interrompili prima di eseguire la sincroniz zazione finale dei dati. Per istruzioni, consultate la documentazione dell'appl icazione o i processi interni per interrompere le attività di lettura e scrittura. Ad esempio,	Proprietario dell'app, sviluppat ore dell'app

Operazione	Description	Competenze richieste
	consultate Avvio o arresto del server Web (IIS 8) (documentazione Microsoft) o Gestione dei servizi di sistema con systemctl (documentazione Red Hat).	
Esegui il trasferimento finale dei dati.	1. Nello strumento di migrazione, eseguite manualmente un'attività o un processo finale di trasferimento dei dati per sincronizzare il file system di destinazione con l'SFS di origine. Per istruzioni, consultate Avvio dell' DataSync attività o consultate la documentazione dello strumento di migrazione di terze parti selezionato. 2. Attendi il completamento dell'operazione di trasferimento dei dati. Per ulteriori informazioni, consulta Monitoraggio AWS DataSync dell'attività con Amazon CloudWatch e Monitoraggio delle DataSync attività dalla riga di comando.	Ingegnere addetto alla migrazione, responsabile della migrazione

Operazione	Description	Competenze richieste
Convalida il trasferimento dei dati.	Se lo utilizzi AWS DataSync, procedi come segue per convalidare il trasferimento finale dei dati completato con successo: 1. Nella AWS DataSync console, prendi nota dell'operazione e dell'ID di esecuzione, ad esempio `task-0000-exec-1111` . 2. Passa alla sezione Task Logging dell' DataSync attività. 3. Scegli il link del gruppo di CloudWatch log. 4. Nei log, cerca l'operazione e l'ID di esecuzione. 5. Prendi nota di eventuali errori di trasferimento. Per ulteriori informazioni, consulta Errori comuni nella DataSync documentazione. 6. Convalida quanto segue: • Confronta gli elenchi di file dall'origine e dalla destinazione SFSs per confermare che tutti i dati siano stati trasferiti • Confronta le autorizza zioni di accesso ai file tra	Ingegnere addetto alla migrazione, responsabile della migrazione

Operazione	Description	Competenze richieste
	l'origine e la destinazione SFSs. Se utilizzi uno strumento di terze parti, consulta le istruzioni per la convalida del trasferimento dei dati nella documentazione dello strumento di migrazione selezionato.	

Convalida

Operazione	Description	Competenze richieste
Rimontare il file system e convalidare la funzione e le prestazioni dell'applicazione.	1. Se i server dipendenti sono stati migrati in questa ondata, nel file SFS-Wave-Plan-Workbook, nel foglio di lavoro Remount-Summary, inserisci il nuovo indirizzo IP del server nella colonna Nuovo indirizzo IP del server (F). 2. Su tutti i server, aggiorna il punto di montaggio per il file system dal vecchio percorso al nuovo percorso. Utilizzate il runbook della vostra organizzazione per il rimontaggio discusso in precedenza nella fase di preparazione.	Amministratore di sistema AWS, proprietario dell'app

Operazione	Description	Competenze richieste
	3. Verificate che il file system sia montato correttamente e sia accessibile controllando i supporti e verificando che i file siano presenti. Il team dell'infrastruttura in genere esegue queste attività. 4. Riavvia le applicazioni e coinvolgi i proprietari delle applicazioni o il team di controllo qualità per completare i test funzionali e prestazionali sull'applicazione, in base alle esigenze dell'applicazione.	

risoluzione dei problemi

Problema	Soluzione
I valori delle celle in Microsoft Excel non vengono aggiornati.	Copia le formule nelle righe di esempio trascinando la maniglia di riempimento. Per ulteriori informazioni, consulta le istruzioni per Windows o per Mac (sito Web Microsoft Support).

Risorse correlate

AWS documentazione

- AWS DataSync documentazione
- Documentazione Amazon EFS
- FSx Documentazione Amazon
- Grandi migrazioni verso Cloud AWS

- Guida per migrazioni di AWS grandi dimensioni
- Guida al portfolio per AWS migrazioni di grandi dimensioni

Risoluzione dei problemi

- Risoluzione dei problemi AWS DataSync
- Risoluzione dei problemi di Amazon EFS
- Risoluzione dei problemi FSx di Amazon per Windows File Server
- Risoluzione dei problemi di Amazon FSx for NetApp ONTAP

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Esegui la migrazione di un database Oracle ad Amazon RDS for Oracle utilizzando gli adattatori flat file GoldenGate Oracle

Dhairya Jindani e Baji Shaik, Amazon Web Services

Riepilogo

Oracle GoldenGate è un servizio di acquisizione e replica dei dati in tempo reale per database e ambienti IT eterogenei. Tuttavia, questo servizio attualmente non supporta Amazon Relational Database Service (Amazon RDS) per Oracle. Per un elenco dei database supportati, consulta Oracle GoldenGate for Heterogeneous Databases (documentazione Oracle). Questo modello descrive come utilizzare gli adattatori GoldenGate flat file Oracle GoldenGate e Oracle per generare file flat dal database Oracle di origine, che può essere locale o su un'istanza Amazon Elastic Compute Cloud (Amazon EC2). È quindi possibile importare tali file flat in un'istanza di database Amazon RDS for Oracle.

In questo modello, si utilizza Oracle GoldenGate per estrarre i file trail dal database Oracle di origine. Il data pump copia i file trail su un server di integrazione, che è un' EC2 istanza Amazon. Sul server di integrazione, Oracle GoldenGate utilizza l'adattatore flat file per generare una serie di file flat sequenziali basati sull'acquisizione dei dati transazionali dei file trail. Oracle GoldenGate formatta i dati come valori separati da delimitatori o valori delimitati dalla lunghezza. Si utilizza quindi Oracle SQL*Loader per importare i file flat nell'istanza di database Amazon RDS for Oracle di destinazione.

Destinatari

Questo modello è destinato a coloro che hanno esperienza e conoscenza degli elementi costitutivi fondamentali GoldenGate di un Oracle. Per ulteriori informazioni, vedere Panoramica dell' GoldenGate architettura Oracle (documentazione Oracle).

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS.
- Una GoldenGate licenza Oracle.
- Una licenza separata per un GoldenGate adattatore Oracle.
- Un database Oracle di origine, in esecuzione in locale o su un' EC2 istanza Amazon.
- Un'istanza Amazon EC2 Linux utilizzata come server di integrazione. Per ulteriori informazioni, consulta Guida introduttiva alle istanze Amazon EC2 Linux (EC2 documentazione Amazon).

- Un'istanza di database Amazon RDS for Oracle di destinazione. Per ulteriori informazioni, consulta Creazione di un'istanza DB Oracle (documentazione Amazon RDS).

Versioni del prodotto

- Oracle Database Enterprise Edition versione 10g, 11g, 12c o successiva
- Oracle GoldenGate versione 12.2.0.1.1 o successiva

Architecture

Stack tecnologico di origine

Un database Oracle (locale o su un' EC2 istanza Amazon)

Stack tecnologico Target

Amazon RDS per Oracle

Architettura di origine e destinazione

1. Oracle GoldenGate estrae le tracce dai log del database di origine.
2. La data pump estrae le tracce e le migra su un server di integrazione.
3. L'adattatore di file GoldenGate flat di Oracle legge gli itinerari, le definizioni di origine e i parametri di estrazione.
4. Si esce dall'estrazione, che genera un control file e file di dati flat.
5. È possibile migrare i file di dati flat su un'istanza di database Amazon RDS for Oracle in. Cloud AWS

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon Relational Database Service (Amazon RDS) per Oracle ti aiuta a configurare, gestire e scalare un database relazionale Oracle in. Cloud AWS

Altri servizi

- Oracle GoldenGate è un servizio che consente di replicare, filtrare e trasformare i dati da un database a un altro database eterogeneo o a un'altra topologia di destinazione, come i file flat.
- Gli GoldenGate application adapter Oracle consentono GoldenGate a Oracle di produrre una serie di file flat sequenziali e file di controllo a partire dai dati transazionali acquisiti nei file trail di un database di origine. Questi adattatori sono ampiamente utilizzati per le operazioni di estrazione, trasformazione e caricamento (ETL) in applicazioni di data warehouse e applicazioni proprietarie o legacy. Oracle GoldenGate esegue questa acquisizione e la applica quasi in tempo reale su database, piattaforme e sistemi operativi eterogenei. Gli adattatori supportano diversi formati per i file di output, come CSV o Apache Parquet. È possibile caricare questi file generati per caricare i dati in diversi database eterogenei.

Epiche

Configura Oracle GoldenGate sul server del database di origine

Operazione	Description	Competenze richieste
Scarica Oracle GoldenGate.	Sul server del database di origine, scarica la GoldenGate versione Oracle 12.2.0.1.1 o successiva. Per istruzioni, vedere Download di Oracle GoldenGate (documentazione Oracle).	DBA
Installa Oracle GoldenGate.	Per istruzioni, vedere Installazione di Oracle GoldenGate (documentazione Oracle).	DBA
Configura Oracle GoldenGate.	Per istruzioni, vedere Preparazione del database per Oracle GoldenGate (documentazione Oracle).	DBA

Configura Oracle GoldenGate sul server di integrazione

Operazione	Description	Competenze richieste
Scarica Oracle GoldenGate.	Sul server di integrazione, scarica la GoldenGate versione Oracle 12.2.0.1.1 o successiva. Per istruzioni, vedere Download di Oracle GoldenGate (documentazione Oracle).	DBA
Installa Oracle GoldenGate.	Crea directory, configura il processo di gestione e crea il `defgen` file per un ambiente eterogeneo. Per istruzioni, vedere Installazione di Oracle GoldenGate (documentazione Oracle).	DBA

Modifica la configurazione di acquisizione GoldenGate dei dati Oracle

Operazione	Description	Competenze richieste
Preparare gli GoldenGate adattatori Oracle.	Sul server di integrazione, configura il software dell' GoldenGate adattatore Oracle. Esegui questa operazione: 1. Da Oracle Software Delivery Cloud, scarica ggs_Adapters_Linux _x64.zip. 2. Decomprimi ggs_Adapt ers_Linux_x64.zip.	DBA

Operazione	Description	Competenze richieste
	3. Eseguite il seguente comando per installare gli adattatori: `tar -xvf ggs_Adapters_Linux_x64.tar`	
Configura la pompa dati.	Sul server di origine, configura la pompa di dati per trasferire il file trail dal server di origine al server di integrazione. Crea il file dei parametri della pompa dati e la directory dei file trail. Per istruzioni, vedere Configurazione del Flat File Adapter (documentazione Oracle).	DBA

Genera e migra i file flat

Operazione	Description	Competenze richieste
Genera i file flat.	Crea il file di estrazione e il file di controllo, quindi avvia il processo di estrazione sul server di integrazione. Questo estrae le modifiche al database e scrive il database di origine nei file flat. Per istruzioni, vedere Using the Flat File Adapter (documentazione Oracle).	DBA
Carica i file flat nel database di destinazione.	Carica i file flat nell'istanza di database Amazon RDS for	DBA

Operazione	Description	Competenze richieste
	Oracle di destinazione. Per ulteriori informazioni, consulta Importazione tramite Oracle SQL*Loader (documentazione Amazon RDS).	

risoluzione dei problemi

Problema	Soluzione
L'adattatore di file GoldenGate flat Oracle genera un errore.	Per una descrizione degli errori dell'adattatore, vedere Localizzazione dei messaggi di errore (documentazione Oracle). Per istruzioni sulla risoluzione dei problemi, vedere Risoluzione dei problemi del Flat File Adapter (documentazione Oracle).

Risorse correlate

- Installazione di Oracle GoldenGate (documentazione Oracle)
- Configurazione di Oracle GoldenGate (documentazione Oracle)
- Informazioni sugli GoldenGate adattatori Oracle (documentazione Oracle)
- Configurazione del Flat File Adapter (documentazione Oracle)

Modifica le applicazioni Python e Perl per supportare la migrazione dei database da Microsoft SQL Server a Amazon Aurora PostgreSQL Compatible Edition

Dwarika Patra e Deepesh Jayaprakash, Amazon Web Services

Riepilogo

Questo modello descrive le modifiche ai repository delle applicazioni che potrebbero essere necessarie durante la migrazione dei database da Microsoft SQL Server a Amazon Aurora PostgreSQL Compatible Edition. Il modello presuppone che queste applicazioni siano basate su Python o su Perl e fornisce istruzioni separate per questi linguaggi di scripting.

La migrazione dei database di SQL Server verso la compatibilità con Aurora PostgreSQL comporta la conversione dello schema, la conversione degli oggetti del database, la migrazione dei dati e il caricamento dei dati. A causa delle differenze tra PostgreSQL e SQL Server (relative ai tipi di dati, agli oggetti di connessione, alla sintassi e alla logica), l'attività di migrazione più difficile consiste nell'apportare le modifiche necessarie alla base di codice in modo che funzioni correttamente con PostgreSQL.

Per un'applicazione basata su Python, gli oggetti e le classi di connessione sono sparsi in tutto il sistema. Inoltre, la base di codice Python potrebbe utilizzare più librerie per connettersi al database. Se l'interfaccia di connessione al database cambia, anche gli oggetti che eseguono le query in linea dell'applicazione richiedono modifiche.

Per un'applicazione basata su Perl, le modifiche riguardano oggetti di connessione, driver di connessione al database, istruzioni SQL in linea statiche e dinamiche e il modo in cui l'applicazione gestisce query DML dinamiche complesse e set di risultati.

Quando esegui la migrazione della tua applicazione, puoi anche prendere in considerazione possibili miglioramenti su AWS, come la sostituzione del server FTP con l'accesso ad Amazon Simple Storage Service (Amazon S3).

Il processo di migrazione delle applicazioni comporta le seguenti sfide:

- Oggetti di connessione. Se gli oggetti di connessione sono sparsi nel codice con più librerie e chiamate di funzioni, potrebbe essere necessario trovare un modo generalizzato per modificarli per supportare PostgreSQL.
- Gestione degli errori o delle eccezioni durante il recupero o l'aggiornamento dei record. Se sul database sono presenti operazioni condizionali di creazione, lettura, aggiornamento ed

eliminazione (CRUD) che restituiscono variabili, set di risultati o frame di dati, eventuali errori o eccezioni potrebbero causare errori di applicazione con effetti a cascata. Queste devono essere gestite con cura con convalide e punti di salvataggio adeguati. Uno di questi punti di salvataggio consiste nel richiamare query SQL in linea di grandi dimensioni o oggetti di database all'interno di blocchi. `BEGIN...EXCEPTION...END`

- Controllo delle transazioni e loro convalida. Questi includono commit e rollback manuali e automatici. Il driver PostgreSQL per Perl richiede di impostare sempre in modo esplicito l'attributo auto-commit.
- Gestione di query SQL dinamiche. Ciò richiede una conoscenza approfondita della logica delle query e dei test iterativi per garantire che le query funzionino come previsto.
- Prestazioni. È necessario assicurarsi che le modifiche al codice non comportino un peggioramento delle prestazioni dell'applicazione.

Questo modello spiega in dettaglio il processo di conversione.

Prerequisiti e limitazioni

Prerequisiti

- Conoscenza pratica della sintassi Python e Perl.
- Competenze di base in SQL Server e PostgreSQL.
- Comprensione dell'architettura applicativa esistente.
- Accesso al codice dell'applicazione, al database SQL Server e al database PostgreSQL.
- Accesso all'ambiente di sviluppo Windows o Linux (o altro Unix) con credenziali per lo sviluppo, il test e la convalida delle modifiche alle applicazioni.
- Per un'applicazione basata su Python, le librerie Python standard che l'applicazione potrebbe richiedere, come Pandas per gestire i frame di dati e psycopg2 per le connessioni al database. SQLAlchemy
- Per un'applicazione basata su Perl, sono necessari pacchetti Perl con librerie o moduli dipendenti. Il modulo Comprehensive Perl Archive Network (CPAN) può supportare la maggior parte dei requisiti delle applicazioni.
- Tutte le librerie o i moduli personalizzati dipendenti richiesti.
- Credenziali del database per l'accesso in lettura a SQL Server e read/write l'accesso ad Aurora.
- PostgreSQL per convalidare ed eseguire il debug delle modifiche alle applicazioni con servizi e utenti.

- Accesso a strumenti di sviluppo durante la migrazione delle applicazioni come Visual Studio Code, Sublime Text o pgAdmin.

Limitazioni

- Alcune versioni, moduli, librerie e pacchetti di Python o Perl non sono compatibili con l'ambiente cloud.
- Alcune librerie e framework di terze parti utilizzati per SQL Server non possono essere sostituiti per supportare la migrazione PostgreSQL.
- Le variazioni delle prestazioni potrebbero richiedere modifiche all'applicazione, alle query Transact-SQL (T-SQL) in linea, alle funzioni del database e alle stored procedure.
- PostgreSQL supporta nomi minuscoli per nomi di tabelle, nomi di colonne e altri oggetti di database.
- Alcuni tipi di dati, come le colonne UUID, vengono memorizzati solo in lettere minuscole. Le applicazioni Python e Perl devono gestire tali differenze di casi.
- Le differenze di codifica dei caratteri devono essere gestite con il tipo di dati corretto per le colonne di testo corrispondenti nel database PostgreSQL.

Versioni del prodotto

- Python 3.6 o versione successiva (usa la versione che supporta il tuo sistema operativo)
- Perl 5.8.3 o versione successiva (usa la versione che supporta il tuo sistema operativo)
- Aurora PostgreSQL Compatible Edition 4.2 o versione successiva (vedi dettagli)

Architecture

Stack tecnologico di origine

- Linguaggio di scripting (programmazione di applicazioni): Python 2.7 o successivo o Perl 5.8
- Database: Microsoft SQL Server versione 13
- Sistema operativo: Red Hat Enterprise Linux (RHEL) 7

Stack tecnologico Target

- Linguaggio di scripting (programmazione di applicazioni): Python 3.6 o successivo o Perl 5.8 o successivo
- Database: Aurora PostgreSQL compatibile 4.2
- Sistema operativo: RHEL 7

Architettura di migrazione

Tools (Strumenti)

Servizi e strumenti AWS

- Aurora PostgreSQL—Compatible Edition è un motore di database relazionale completamente gestito, compatibile con PostgreSQL e conforme ad ACID che combina la velocità e l'affidabilità dei database commerciali di fascia alta con l'economicità dei database open source. Aurora PostgreSQL è un sostituto immediato di PostgreSQL e semplifica e rende più conveniente configurare, utilizzare e scalare le implementazioni PostgreSQL nuove ed esistenti.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire con i servizi AWS utilizzando i comandi nella shell della riga di comando.

Altri strumenti

- Librerie di connessione al database Python e PostgressSQL come psycopg2 e SQLAlchemy
- Perl e i suoi moduli DBI
- Terminale interattivo PostgreSQL (psql)

Epiche

Migra il tuo repository di applicazioni a PostgreSQL: passaggi di alto livello

Operazione	Description	Competenze richieste
Segui questi passaggi di conversione del codice per migrare la tua applicazione a PostgreSQL.	1. Imposta driver e librerie ODBC specifici del database per PostgreSQL. Ad esempio, puoi usare uno	Sviluppatore di app

Operazione	Description	Competenze richieste
	dei moduli CPAN per Perl e pyodbc, psycopg2 o per Python. SQLAlchemy 2. Converti gli oggetti del database utilizzando queste librerie per connetterti a Aurora PostgreSQL compatibile. 3. Applica modifiche al codice nei moduli applicativi esistenti per ottenere istruzioni T-SQL compatibili. 4. Riscrivi le chiamate di funzione specifiche del database e le stored procedure nel codice dell'applicazione. 5. Gestisci le modifiche alle variabili dell'applicazione e ai relativi tipi di dati utilizzati per le query SQL in linea. 6. Gestisci funzioni specifiche del database incompatibili. 7. end-to-endTest completo del codice applicativo convertito per la migrazione del database. 8. Confronta i risultati di Microsoft SQL Server con l'applicazione che hai migrato a PostgreSQL. 9. Esegui il benchmarking delle prestazioni delle	

Operazione	Description	Competenze richieste
	applicazioni tra Microsoft SQL Server e PostgreSQL. 10. Rivedi le stored procedure o le istruzioni T-SQL in linea richiamate dall'applicazione per migliorare le prestazioni. I seguenti poemi epici forniscono istruzioni dettagliate per alcune di queste attività di conversione per applicazioni Python e Perl.	

Operazione	Description	Competenze richieste
Usa una lista di controllo per ogni fase della migrazione.	Aggiungi quanto segue alla tua lista di controllo per ogni fase della migrazione delle applicazioni, inclusa la fase finale: • Consulta la documentazione di PostgreSQL per assicurarti che tutte le modifiche siano compatibili con lo standard PostgreSQL. • Controlla i valori interi e mobili per le colonne. • Identifica il numero di righe inserite, aggiornate ed estratte, insieme ai nomi e ai timbri delle colonne. date/time È possibile utilizzare un'utilità diff o scrivere uno script per automatizzare questi controlli. • Completa i controlli delle prestazioni per istruzioni SQL in linea di grandi dimensioni e verifica le prestazioni complessive dell'applicazione. • Verificate la corretta gestione degli errori nelle operazioni del database e la corretta uscita dal programma utilizzando più blocchi. try/catch	Sviluppatore di app

Operazione	Description	Competenze richieste
	• Verificate che i processi di registrazione siano corretti.	

Analizza e aggiorna la tua applicazione — Python code base

Operazione	Description	Competenze richieste
Analizza la tua base di codice Python esistente.	L'analisi dovrebbe includere quanto segue per facilitare il processo di migrazione delle applicazioni: • Identifica tutti gli oggetti di connessione nel codice. • Identifica tutte le query SQL in linea incompatibili (come le istruzioni T-SQL e le stored procedure) e analizza le modifiche necessarie. • Consulta la documentazione relativa al codice e monitora il flusso di controllo per comprendere la funzional ità del codice. Ciò sarà utile in seguito, quando testerai l'applicazione per il confronto delle prestazioni o del carico. • Comprendete lo scopo dell'applicazione in modo da poterla testare in modo efficace dopo la conversio ne del database. La maggior parte delle applicazioni	Sviluppatore di app

Operazione	Description	Competenze richieste
	Python candidate alla conversione con migrazioni di database sono feed che caricano dati da altre fonti in tabelle di database o estrattori che recuperano i dati dalle tabelle e li trasformano in diversi formati di output (come CSV, JSON o file flat) adatti per creare report o effettuare chiamate API per eseguire convalide.	

Operazione	Description	Competenze richieste

Converti le connessioni al database per supportare PostgreSQL.	La maggior parte delle applicazioni Python utilizza la libreria pyodbc per connetter si ai database di SQL Server come segue.
	
	`import pyodbc`
	`....`
	`try:`
	`conn_string = "Driver=ODBC Driver 17 for SQL Server;UID={};PWD={};Server={};Database={}".format (conn_user, conn_password, conn_server, conn_database)`
	`conn = pyodbc.connect(conn_string)`
	`cur = conn.cursor()`
	`result = cur.execute(query_string)`
	`for row in result:`
	`print (row)`
	`except Exception as e:`
	`print(str(e))`
	
	Converti la connessione al database per supportare PostgreSQL come segue.
	
	`import pyodbc`
	`import psycopg2`
	`....`
	`try:`
		Sviluppatore di app

Operazione	Description	Competenze richieste
	`conn_string = 'postgresql+psycopg2://'+ conn_user+':'+conn_password+'@'+conn_server+'/'+conn_database conn = pyodbc.connect(conn_string, connect_args={'options':'-csearch_path=dbo'}) cur = conn.cursor() result = cur.execute(query_string) for row in result: print (row) except Exception as e: print(str(e))`	

Operazione	Description	Competenze richieste

Cambia le query SQL in linea in PostgreSQL.	Converti le tue query SQL in linea in un formato compatibile con PostgreSQL. Ad esempio, la seguente query di SQL Server recupera una stringa da una tabella.
	
	`dtype = "type1"`
	`stm = ‘"SELECT TOP 1 searchcode FROM TypesTable (NOLOCK) WHERE code="’ + "’" + str(dtype) + "’"`
	`# For Microsoft SQL Server Database Connection`
	`engine = create_engine(‘mssql+pyodbc:///?odbc_connect=%s’ % urllib.parse.quote_plus(conn_string), connect_args={‘connect_timeout’:login_timeout})`
	`conn = engine_connect()`
	`rs = conn.execute(stm)`
	`for row in rs:`
	`print(row)`
	
	Dopo la conversione, la query SQL in linea compatibile con PostgreSQL ha il seguente aspetto.
	
	`dtype = "type1"`
	`stm = ‘"SELECT searchcode FROM TypesTable`
		Sviluppatore di app

Operazione	Description	Competenze richieste

	`WHERE code="’ + "’" + str(dtype) + "’ LIMIT 1"`
	`# For PostgreSQL Database Connection`
	`engine = create_engine(‘postgres+psycopg2://%s’ %conn_string, connect_args={‘connect_timeout’:login_timeout})`
	`conn = engine.connect()`
	`rs = conn.execute(stm)`
	`for row in rs:`
	`  print(row)`	

Operazione	Description	Competenze richieste

Gestisci le query SQL dinamiche.	L'SQL dinamico può essere presente in uno script o in più script Python. Gli esempi precedenti hanno mostrato come utilizzare la funzione di sostituzione delle stringhe di Python per inserire variabili per la costruzione di query SQL dinamiche. Un approccio alternativo consiste nell'aggi ungere la stringa di query con variabili laddove applicabile.
	
	Nell'esempio seguente, la stringa di query viene costruita al volo in base ai valori restituit i da una funzione.
	
	`query = '"SELECT id from equity e join issues i on e.permId=i.permId where e.id'"`
	`query += get_id_fi lter(ids) + " e.id is NOT NULL`
	
	Questi tipi di interrogazioni dinamiche sono molto comuni durante la migrazione delle applicazioni. Segui questi passaggi per gestire le query dinamiche:
	
	• Controllate la sintassi generale (ad esempio, la sintassi per
		Sviluppatore di app

Operazione	Description	Competenze richieste
	l'`SELECT`istruzione con una `JOIN` clausola). • Verifica tutte le variabili o i nomi di colonna utilizzati nella query, ad esempio e. `i id` • Controllate le funzioni, gli argomenti e i valori restituiti utilizzati nella query (ad esempio, `get_id_filter` e il relativo argomento`ids`).	

Operazione	Description	Competenze richieste
Gestisci set di risultati, variabili e frame di dati.	Per Microsoft SQL Server, si utilizzano metodi Python come `fetchone()` o `fetchall()` per recuperare e il set di risultati dal database. È inoltre possibile utilizzare `fetchmany(size)` e specificare il numero di record da restituire dal set di risultati. A tale scopo, è possibile utilizzare l'oggetto di connessione pyodbc come illustrato nell'esempio seguente. pyodbc (Microsoft SQL Server) `import pyodbc server = 'tcp:myserver.database.windows.net' database = 'exampledb' username = 'exampleusername' password = 'examplepassword' conn = pyodbc.connect('DRIVER={ODBC Driver 17 for SQL Server};SERVER='+server+';DATABASE='+database+';UID='+username+';PWD='+ password) cursor = conn.cursor() cursor.execute("SELECT * FROM ITEMS")`	Sviluppatore di app

Operazione	Description	Competenze richieste
	`row = cursor.fe tchone() while row: print(row[0]) row = cursor.fe tchone()`	
	In Aurora, per eseguire attività simili come la connessione a PostgreSQL e il recupero dei set di risultati, puoi usare psycopg2 o. SQLAlchem y Queste librerie Python forniscono il modulo di connessione e l'oggetto cursore da attraversare tra i record del database PostgreSQL, come mostrato nell'esempio seguente. psycopg2 (compatibile con Aurora PostgreSQL) `import psycopg2 query = "SELECT * FROM ITEMS;" //Initialize variables host=dbname=user= password=port=sslm ode=connect_timeou t="" connstring = "host='{h ost}' dbname='{ dbname}' user='{user}' \ password='{passw ord}'port='{port}'`	

Operazione	Description	Competenze richieste
	".format(host=host ,dbname=dbname,\ user=user,password= password,port=port) conn = psycopg2. connect(connstring) cursor = conn.cursor() cursor.execute(query) column_names = [column[0] for column in cursor.description] print("Column Names: ", column_names) print("Column values: " for row in cursor: print("itemid :", row[0]) print("itemdescrpt ion :", row[1]) print("it emprice :", row[3]))	
	SQLAlchemy (Compatibile con Aurora PostgreSQL) from sqlalchemy import create_engine from pandas import DataFrame conn_string = 'postgres ql://core:database @localhost:5432/ex ampledatabase' engine = create_en gine(conn_string) conn = engine.co nnect() dataid = 1001	

Operazione	Description	Competenze richieste
	`result = conn.exec ute("SELECT * FROM ITEMS") df = DataFrame (result.fetchall()) df.columns = result.ke ys() df = pd.DataFrame() engine.connect() df = pd.read_sql_query( sql_query, engine, coerce_float=False) print("df=", df)`	

Operazione	Description	Competenze richieste
Testa la tua applicazione durante e dopo la migrazione.	Il test dell'applicazione Python migrata è un processo continuo. Poiché la migrazione include modifiche agli oggetti di connessione (psycopg2 o SQLAlchemy), gestione degli errori, nuove funzionalità (frame di dati), modifiche SQL in linea, funzionalità di copia in blocco (`bcp`anzi ché`COPY`) e modifiche simili, deve essere testata attentamente durante e dopo la migrazione dell'applicazione. Controlla: • Condizioni e gestione degli errori • Eventuali mancate corrispondenze tra i record dopo la migrazione • Aggiornamenti o eliminazioni dei record • Tempo necessario per eseguire l'applicazione	Sviluppatore di app

Analizza e aggiorna la tua applicazione: codice base Perl

Operazione	Description	Competenze richieste
Analizza la tua base di codice Perl esistente.	L'analisi dovrebbe includere quanto segue per facilitare il processo di migrazione delle applicazioni. È necessario identificare:	Sviluppatore di app

Operazione	Description	Competenze richieste
	• Qualsiasi codice INI o basato sulla configurazione • Driver Perl standard Open Database Connectiv ity (ODBC) specifici del database o driver personali zzati • Modifiche al codice richieste per le query in linea e T-SQL • Interazioni tra vari moduli Perl (ad esempio, un singolo oggetto di connessione ODBC in Perl chiamato o utilizzato da più componenti funzionali) • Gestione dei set di dati e dei set di risultati • Librerie Perl esterne e dipendenti • Tutte APIs quelle utilizzate nell'applicazione • Compatibilità delle versioni di Perl e compatibilità dei driver con Aurora PostgreSQL	

Operazione	Description	Competenze richieste

Converti le connessioni dall'applicazione Perl e dal modulo DBI per supportare PostgreSQL.	Le applicazioni basate su Perl utilizzano generalmente il modulo Perl DBI, che è un modulo di accesso al database standard per il linguaggio di programmazione Perl. È possibile utilizzare lo stesso modulo DBI con driver diversi per SQL Server e PostgreSQL.
	
	Per ulteriori informazioni sui moduli Perl richiesti, sulle installazioni e altre istruzioni, consultate la documentazione di DBD: :Pg. L'esempio seguente si connette a Aurora, compatibile con PostgreSQL all'indirizzo. `examplete st-aurorapg-databa se.cluster-samplec lusture.us-east-.r ds.amazonaws.com`
	
	`#!/usr/bin/perl`
	`use DBI;`
	`use strict;`
	`my $driver = "Pg";`
	`my $hostname = "examplet`
	`est-aurorapg-datab`
	`ase-sampleclusture`
	`.us-east.rds.amazo`
	`naws.com"`
	`my $dsn = "DBI:$driver:`
	` dbname = $hostname`
	`;host = 127.0.0.1;port`
	` = 5432";`
		Sviluppatore di app

Operazione	Description	Competenze richieste

	`my $username = "postgres";`
	`my $password = "pass123";`
	`$dbh = DBI->connect("dbi:Pg:dbname=$hostname;host=$host;port=$port;options=$options",`
	`$username,`
	`$password,`
	`{AutoCommit => 0, RaiseError => 1, PrintError => 0}`
	`);`	

Operazione	Description	Competenze richieste
Cambia le query SQL in linea in PostgreSQL.	L'applicazione potrebbe avere query SQL in linea con`SELECT`, `DELETE``UPDATE`, e istruzioni simili che includono clausole di query non supportate da PostgreSQL. Ad esempio, parole chiave di query come `TOP` e `NOLOCK` non sono supportate in PostgreSQL. Gli esempi seguenti mostrano come è possibile gestire le variabili `TOP` `NOLOCK` booleane e.  In SQL Server:  `$sqlStr = $sqlStr . "WHERE a.student_id in (SELECT TOP $numofRecords c_student_id \ FROM active_student_record b WITH (NOLOCK) \ INNER JOIN student_contributor c WITH (NOLOCK) on c.contributor_id = b.c_st)`  Per PostgreSQL, converti in:  `$sqlStr = $sqlStr . "WHERE a.student_id in (SELECT TOP $numofRecords c_student_id \ FROM active_student_record b INNER JOIN`	Sviluppatore di app

Operazione	Description	Competenze richieste
	student_contributor c \ on c.contributor_id = b.c_student_contr_id WHERE b_current_1 is true \ LIMIT $numofRecords)"	

Operazione	Description	Competenze richieste
Gestisci query SQL dinamiche e variabili Perl.	Le query SQL dinamiche sono istruzioni SQL create in fase di esecuzione dell'applicazione. Queste query vengono create dinamicamente durante l'esecuzione dell'applicazione, a seconda di determinate condizioni, in modo che il testo completo della query non sia noto fino all'esecuzione. Un esempio è un'applicazione di analisi finanziaria che analizza quotidianamente le prime 10 azioni e queste azioni cambiano ogni giorno. Le tabelle SQL vengono create in base alle migliori prestazioni e i valori sono noti solo in fase di esecuzione. Supponiamo che le query SQL in linea per questo esempio vengano passate a una funzione wrapper per ottenere i risultati impostati in una variabile e che quindi una variabile utilizzi una condizion e per determinare se la tabella esiste: • Se la tabella esiste, non crearla; esegui alcune elaborazioni.	Sviluppatore di app

Operazione	Description	Competenze richieste

	• Se la tabella non esiste, creala ed esegui anche qualche elaborazione.
	
	Ecco un esempio di gestione delle variabili, seguito dalle query SQL Server e PostgreSQL per questo caso d'uso.
	
	`my $tableexists = db_read( arg 1, $sql_qry, undef, 'writer'); my $table_already_exists = $tableexists->[0]{table_exists}; if ($table_already_exists){ # do some thing } else { # do something else }`
	
	SQL Server:
	
	`my $sql_qry = "SELECT OBJECT_ID('$backendTable', 'U') table_exists", undef, 'writer') ";`
	
	PostgreSQL:
	
	`my $sql_qry = "SELECT TO_REGCLASS('$backendTable', 'U')`	

Operazione	Description	Competenze richieste

	table_exists", undef, 'writer')";
	
	L'esempio seguente utilizza una variabile Perl in SQL in linea, che esegue un'`SELECT`istruzione con `JOIN` a per recuperare la chiave primaria della tabella e la posizione della colonna chiave.
	
	SQL Server:
	
	`my $sql_qry = "SELECT column_name', character_maxi mum_length \ FROM INFORMATION_SCHEMA .COLUMNS \ WHERE TABLE_SCH EMA='$example_sche maInfo' \ AND TABLE_NAME='$examp le_table' \ AND DATA_TYPE IN ('varchar','nvarch ar');";`
	
	PostgreSQL:
	
	`my $sql_qry = "SELECT c1.column_name, c1.ordinal_position \ FROM information_schema .key_column_usage AS c LEFT \`	

Operazione	Description	Competenze richieste
	`JOIN information_schema .table_constraints AS t1 \ ON t1.constraint_name = c1.constraint_name \ WHERE t1.table_name = $example_schemaInf o'.'$example_table' \ AND t1.constraint_type = 'PRIMARY KEY' ;";`	

Apporta ulteriori modifiche alla tua applicazione basata su Perl o Python per supportare PostgreSQL

Operazione	Description	Competenze richieste
Converti costrutti SQL Server aggiuntivi in PostgreSQL.	Le seguenti modifiche si applicano a tutte le applicazioni, indipendentemente dal linguaggio di programmazione. • Qualifica gli oggetti di database utilizzati dall'applicazione con nomi di schema nuovi e appropriati. • Gestisci gli operatori LIKE per la corrispondenza con distinzione tra maiuscole e minuscole con la funzione di confronto di PostgreSQL. • Gestisci funzioni specifiche del database non supportate come `DATEDIFF`,,, e operatori. `DATEADD` `GETDATE` `CONVERT` `CAST` Per funzioni equivalenti	Sviluppatore di app

Operazione	Description	Competenze richieste
	compatibili con PostgreSQL, consulta Funzioni SQL native o integrate nella sezione Informazioni aggiuntive. • Gestisci i valori booleani nelle istruzioni di confronto. • Gestisce i valori restituit i dalle funzioni. Questi potrebbero essere set di record, frame di dati, variabili e valori booleani. Gestiscili in base ai requisiti della tua applicazione e per supportare PostgreSQL. • Gestisci blocchi anonimi (ad esempio`BEGIN TRAN`) con nuove funzioni PostgreSQL definite dall'utente. • Converti inserti in blocco per righe. L'equivalente PostgreSQL dell'utilità bulk copy `bcp` () di SQL Server, che viene richiamata dall'interno dell'applicazione, è. `COPY` • Converte gli operatori di concatenazione di colonne. SQL Server utilizza + per la concatenazione di stringhe, ma PostgreSQL utilizza. `||`	

Migliora le prestazioni

Operazione	Description	Competenze richieste
Sfrutta i servizi AWS per migliorare le prestazioni.	Quando esegui la migrazione al cloud AWS, puoi perfezion are la progettazione di applicazioni e database per sfruttare i servizi AWS. Ad esempio, se le query della tua applicazione Python, connessa a un server di database Aurora compatibile con PostgreSQL, richiedono più tempo rispetto alle query originali di Microsoft SQL Server, potresti prendere in considerazione la creazione di un feed di dati storici direttamente in un bucket Amazon Simple Storage Service (Amazon S3) dal server Aurora e utilizzare Amazon Athena Query SQL per generare report e query di dati analitici per i dashboard degli utenti.	Sviluppatore di app, architetto cloud

Risorse correlate

- Perl
- Modulo Perl DBI
- Python
- psycopg2
- SQLAlchemy

- Copia in blocco - PostgreSQL
- Copia in blocco - Microsoft SQL Server
- PostgreSQL
- Lavorare con Amazon Aurora PostgreSQL

Informazioni aggiuntive

Sia Microsoft SQL Server che Aurora PostgreSQL sono compatibili con ANSI SQL. Tuttavia, dovresti comunque essere consapevole di eventuali incompatibilità nella sintassi, nei tipi di dati delle colonne, nelle funzioni native specifiche del database, negli inserimenti in blocco e nella distinzione tra maiuscole e minuscole quando migri la tua applicazione Python o Perl da SQL Server a PostgreSQL.

Le sezioni seguenti forniscono ulteriori informazioni sulle possibili incongruenze.

Confronto dei tipi di dati

Le modifiche ai tipi di dati da SQL Server a PostgreSQL possono portare a differenze significative nei dati risultanti su cui operano le applicazioni. Per un confronto dei tipi di dati, consulta la tabella sul sito Web Sqlines.

Funzioni SQL native o integrate

Il comportamento di alcune funzioni è diverso tra i database SQL Server e PostgreSQL. La tabella seguente fornisce un confronto.

Microsoft SQL Server	Description	PostgreSQL
`CAST`	Converte un valore da un tipo di dati a un altro.	PostgreSQL `type :: operator`
`GETDATE()`	Restituisce la data e l'ora correnti del sistema di database, in un formato. `YYYY-MM-DD hh:mm:ss.mmm`	`CLOCK_TIMESTAMP`
`DATEADD`	Aggiunge un time/date intervallo a una data.	`INTERVAL`espressione

`CONVERT`	Converte un valore in un formato di dati specifico.	`TO_CHAR`
`DATEDIFF`	Restituisce la differenza tra due date.	`DATE_PART`
`TOP`	Limita il numero di righe in un set di `SELECT` risultati.	`LIMIT/FETCH`

Blocchi anonimi

Una query SQL strutturata è organizzata in sezioni quali dichiarazione, eseguibili e gestione delle eccezioni. La tabella seguente confronta le versioni Microsoft SQL Server e PostgreSQL di un semplice blocco anonimo. Per blocchi anonimi complessi, si consiglia di richiamare una funzione di database personalizzata all'interno dell'applicazione.

Microsoft SQL Server

```
my $sql_qry1=
my $sql_qry2 =
my $sqlqry = "BEGIN TRAN
$sql_qry1 $sql_qry2
if @\@error !=0 ROLLBACK
TRAN
else COMIT TRAN";
```

PostgreSQL

```
my $sql_qry1=
my $sql_qry2 =
my $sql_qry = " DO \$\$
BEGIN
$header_sql $content_sql
END
\$\$";
```

Altre differenze

- Inserimenti di righe in blocco: l'equivalente PostgreSQL dell'utilità bcp di Microsoft SQL Server è COPY.
- Sensibilità tra maiuscole e minuscole: i nomi delle colonne fanno distinzione tra maiuscole e minuscole in PostgreSQL, quindi è necessario convertire i nomi delle colonne di SQL Server in lettere minuscole o maiuscole. Questo diventa un fattore quando si estraggono o si confrontano dati o si inseriscono nomi di colonna in set di risultati o variabili. L'esempio seguente identifica le colonne in cui i valori possono essere memorizzati in lettere maiuscole o minuscole.

```
my $sql_qry = "SELECT $record_id FROM $exampleTable WHERE LOWER($record_name) =
 \'failed transaction\'";
```

- Concatenazione: SQL Server utilizza + come operatore per la concatenazione di stringhe, mentre PostgreSQL utilizza. | |
- Convalida: è necessario testare e convalidare le query e le funzioni SQL in linea prima di utilizzarle nel codice dell'applicazione per PostgreSQL.
- Inclusione della libreria ORM: puoi anche cercare di includere o sostituire la libreria di connessione al database esistente con librerie ORM Python come SQLAlchemyPython e PynomODB. Ciò contribuirà a interrogare e manipolare facilmente i dati da un database utilizzando un paradigma orientato agli oggetti.

Modelli di migrazione per carico di lavoro

Argomenti

- IBM
- Microsoft
- N/D
- Open source
- Oracle
- SAP

IBM

- Esegui la migrazione di un database Db2 da Amazon a EC2 Aurora compatibile con MySQL utilizzando AWS DMS
- Esegui la migrazione di Db2 for LUW ad Amazon EC2 utilizzando la spedizione di log per ridurre i tempi di interruzione
- Esegui la migrazione di Db2 per LUW ad Amazon EC2 con disaster recovery ad alta disponibilità
- Esegui la migrazione da IBM Db2 su Amazon a EC2 Aurora PostgreSQL compatibile con AWS DMS e AWS SCT
- Esegui la migrazione da IBM WebSphere Application Server ad Apache Tomcat su Amazon EC2

Microsoft

- Accelera la scoperta e la migrazione dei carichi di lavoro Microsoft su AWS
- Modifica le applicazioni Python e Perl per supportare la migrazione dei database da Microsoft SQL Server a Amazon Aurora PostgreSQL Compatible Edition
- Crea CloudFormation modelli AWS per attività AWS DMS utilizzando Microsoft Excel e Python
- Esportazione di un database Microsoft SQL Server in Amazon S3 utilizzando AWS DMS
- SHA1 Implementa l'hashing per i dati PII durante la migrazione da SQL Server a PostgreSQL
- Acquisisci e migra istanze EC2 Windows in un account AWS Managed Services
- Esegui la migrazione di una coda di messaggistica da Microsoft Azure Service Bus ad Amazon SQS
- Esegui la migrazione di un database Microsoft SQL Server da Amazon EC2 ad Amazon DocumentDB utilizzando AWS DMS
- Esegui la migrazione di un database Microsoft SQL Server su Aurora MySQL utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un'applicazione.NET da Microsoft Azure App Service ad AWS Elastic Beanstalk
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon EC2
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon RDS for SQL Server
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon RDS for SQL Server utilizzando server collegati
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon RDS for SQL Server utilizzando metodi di backup e ripristino nativi
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon Redshift utilizzando AWS DMS
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon Redshift utilizzando gli agenti di estrazione dati AWS SCT
- Esegui la migrazione di un database Microsoft SQL Server locale a Microsoft SQL Server su Amazon EC2 con Linux
- Esegui la migrazione dei dati da Microsoft Azure Blob ad Amazon S3 utilizzando Rclone
- Esegui la migrazione di applicazioni ospitate da IIS su Amazon utilizzando appcmd.exe EC2
- Esegui la migrazione di un database Microsoft SQL Server locale su Amazon EC2 utilizzando Application Migration Service

- Migrazione dei certificati SSL di Windows su un Application Load Balancer utilizzando ACM
- Rehosting dei carichi di lavoro locali nel cloud AWS: lista di controllo per la migrazione
- Risolvi gli errori di connessione dopo la migrazione di Microsoft SQL Server al cloud AWS
- Configura un'infrastruttura Multi-AZ per SQL Server Always On FCI utilizzando Amazon FSx

N/D

- Creare un processo di approvazione per le richieste del firewall durante una migrazione di rehosting a AWS

Open source

- Crea utenti e ruoli delle applicazioni in Aurora, compatibile con PostgreSQL
- Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL con e utilizzando e AWS SCT AWS DMS AWS CLI CloudFormation
- Esegui la migrazione di un database MariaDB locale su Amazon RDS for MariaDB utilizzando strumenti nativi
- Esegui la migrazione di un database MySQL locale su Amazon RDS for MySQL
- Esegui la migrazione di un database MySQL locale su Aurora MySQL
- Esegui la migrazione di un database PostgreSQL locale su Aurora PostgreSQL
- Esegui la migrazione di un database Couchbase Server su Amazon EC2
- Esegui la migrazione da IBM WebSphere Application Server ad Apache Tomcat su Amazon EC2 con Auto Scaling
- Esegui la migrazione da PostgreSQL su Amazon RDS per PostgreSQL utilizzando EC2 pglogical
- Esegui la migrazione di applicazioni Java locali su AWS utilizzando AWS App2Container
- Esegui la migrazione dei database MySQL locali su Aurora MySQL utilizzando Percona, Amazon EFS e Amazon S3 XtraBackup
- Esegui la migrazione di tabelle esterne Oracle verso Amazon Aurora, compatibile con PostgreSQL
- Riavvia automaticamente l'AWS Replication Agent senza disabilitarlo SELinux dopo il riavvio di un server di origine RHEL
- Trasporta i database PostgreSQL tra due istanze DB Amazon RDS utilizzando pg_transport

Oracle

- Converti VARCHAR2 (1) il tipo di dati per Oracle in tipo di dati booleano per Amazon Aurora PostgreSQL
- Emula Oracle DR utilizzando un database globale Aurora compatibile con PostgreSQL
- Migrazione incrementale da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL utilizzando Oracle SQL Developer e AWS SCT
- Carica i file BLOB in formato TEXT utilizzando la codifica dei file in Aurora, compatibile con PostgreSQL
- Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL in modalità SSL utilizzando AWS DMS
- Migrazione di un database Amazon RDS for Oracle su Account AWS un altro database Regione AWS e AWS DMS utilizzo per la replica continua
- Esegui la migrazione di un database Oracle locale su Amazon EC2 utilizzando Oracle Data Pump
- Esegui la migrazione di un database Oracle locale ad Amazon OpenSearch Service utilizzando Logstash
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for MySQL utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for Oracle
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for Oracle utilizzando l'importazione diretta di Oracle Data Pump tramite un collegamento al database
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for Oracle utilizzando Oracle Data Pump
- Esegui la migrazione di un database Oracle locale ad Amazon RDS for PostgreSQL utilizzando un bystander Oracle e AWS DMS
- Esegui la migrazione di un database Oracle locale a Oracle su Amazon EC2
- Esegui la migrazione di un database Oracle da Amazon EC2 ad Amazon RDS for MariaDB utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database Oracle da Amazon EC2 ad Amazon RDS for Oracle utilizzando AWS DMS
- Esegui la migrazione di un database Oracle ad Amazon RDS for Oracle utilizzando gli adattatori flat file GoldenGate Oracle

- Esegui la migrazione di un database Oracle ad Amazon Redshift utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database Oracle ad Aurora PostgreSQL utilizzando AWS DMS e AWS SCT
- Esegui la migrazione di un database Oracle JD Edwards EnterpriseOne su AWS utilizzando Oracle Data Pump e AWS DMS
- Esegui la migrazione di una tabella partizionata Oracle su PostgreSQL utilizzando AWS DMS
- Esegui la migrazione di un PeopleSoft database Oracle su AWS utilizzando AWS DMS
- Esegui la migrazione dei dati da un database Oracle locale ad Aurora PostgreSQL
- Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for MySQL
- Esegui la migrazione da Oracle 8i o 9i ad Amazon RDS for PostgreSQL utilizzando viste materializzate e AWS DMS
- Esegui la migrazione da Oracle 8i o 9i ad Amazon RDS for PostgreSQL utilizzando AWS DMS SharePlex
- Esegui la migrazione da Oracle Database ad Amazon RDS for PostgreSQL utilizzando Oracle GoldenGate
- Esegui la migrazione da Oracle su Amazon EC2 ad Amazon RDS for MySQL utilizzando AWS DMS e AWS SCT
- Esegui la migrazione da Oracle WebLogic ad Apache Tomcat (ToMee) su Amazon ECS
- Migrazione di indici basati su funzioni da Oracle a PostgreSQL
- Migrazione delle applicazioni legacy da Oracle Pro*C a ECPG
- Esegui la migrazione dei valori Oracle CLOB su singole righe in PostgreSQL su AWS
- Esegui la migrazione dei codici di errore del database Oracle a un database compatibile con Amazon Aurora PostgreSQL
- Migrazione delle funzioni native di Oracle su PostgreSQL utilizzando le estensioni
- Esegui la migrazione PeopleSoft da Oracle ad Amazon RDS Custom
- Esegui la migrazione della funzionalità Oracle ROWID a PostgreSQL su AWS
- Migrazione dei pacchetti pragma Oracle SERIALLY_REUSABLE in PostgreSQL
- Migra le colonne virtuali generate da Oracle a PostgreSQL
- Configura la funzionalità Oracle UTL_FILE su Aurora, compatibile con PostgreSQL
- Convalida gli oggetti del database dopo la migrazione da Oracle ad Amazon Aurora PostgreSQL

SAP

- Esegui la migrazione da SAP ASE ad Amazon RDS per SQL Server utilizzando AWS DMS
- Esegui la migrazione di SAP ASE su Amazon Aurora, compatibile EC2 con PostgreSQL utilizzando AWS SCT e AWS DMS

Altri modelli

- Accesso Servizi AWS da IBM z/OS installando AWS CLI
- Valuta le prestazioni delle query per la migrazione dei database SQL Server su MongoDB Atlas su AWS
- Automatizza il failover e il failback tra regioni utilizzando DR Orchestrator Framework
- Automatizza le attività del database nell'edizione SQL Server Express in esecuzione su Amazon EC2 utilizzando Task AWS Lambda Scheduler
- Crea un visualizzatore di file mainframe avanzato nel cloud AWS
- Connect ai dati e ai piani di controllo dell'Application Migration Service tramite una rete privata
- Containerizza i carichi di lavoro mainframe che sono stati modernizzati da Blu Age
- Convertire le query JSON Oracle in SQL del database PostgreSQL
- Convertire la funzionalità temporale Teradata NORMALIZE in Amazon Redshift SQL
- Convertire la funzionalità Teradata RESET WHEN in Amazon Redshift SQL
- Copia le tabelle Amazon DynamoDB tra gli account utilizzando AWS Backup
- Implementa un cluster Cassandra su Amazon EC2 con dati statici privati IPs per evitare il ribilanciamento
- Distribuisci applicazioni multi-stack utilizzando AWS CDK con TypeScript
- Distribuisci istanze di cluster di failover di SQL Server su Amazon e EC2 Amazon FSx utilizzando Terraform
- Emula gli array PL/SQL associativi Oracle in Amazon Aurora PostgreSQL e Amazon RDS per PostgreSQL
- Stima le dimensioni del motore Amazon RDS per un database Oracle utilizzando i report AWR
- Genera approfondimenti sui dati utilizzando Modernizzazione del mainframe AWS Amazon Q in Quick Sight
- Gestisci blocchi anonimi nelle istruzioni SQL dinamiche in Aurora PostgreSQL
- Identifica automaticamente le immagini duplicate dei container durante la migrazione a un repository Amazon ECR
- Configura un cluster di failover di Microsoft SQL Server su Amazon EC2 utilizzando FSx per Windows File Server
- Esegui la migrazione dei carichi di lavoro Apache Cassandra su Amazon Keyspaces utilizzando AWS Glue

- Esegui la migrazione dei carichi di lavoro dei container da Azure Red Hat OpenShift (ARO) a Servizio Red Hat OpenShift su AWS (ROSA)
- Esegui la migrazione da Oracle 8i o 9i ad Amazon RDS for Oracle utilizzando AWS DMS SharePlex
- Esegui la migrazione del gruppo di disponibilità Microsoft SQL Server Always On utilizzando AWS Application Migration Service
- Esegui la migrazione di funzioni e procedure Oracle con più di 100 argomenti a PostgreSQL
- Esegui la migrazione da SAP HANA ad AWS utilizzando SAP HSR con lo stesso nome host
- Esegui la migrazione di SQL Server su AWS utilizzando gruppi di disponibilità distribuiti
- Modernizza e implementa le applicazioni mainframe utilizzando e Terraform AWS Transform
- Modernizza i carichi di lavoro di stampa online mainframe su AWS utilizzando Micro Focus Enterprise Server e LRS VPSX/MFI
- Modernizza la gestione dell'output del mainframe AWS utilizzando Rocket Enterprise Server e LRS X PageCenter
- Modifica le intestazioni HTTP durante la migrazione da F5 a un Application Load Balancer su AWS
- Gestione del failover Multi-AZ per i cluster EMR utilizzando Application Recovery Controller
- Analizza le dipendenze degli oggetti per le migrazioni parziali del database da Oracle a PostgreSQL
- Configura una CI/CD pipeline per la migrazione del database utilizzando Terraform
- Configura il disaster recovery per Oracle JD Edwards con EnterpriseOne AWS Elastic Disaster Recovery
- Semplifica la gestione privata dei certificati utilizzando AWS Private CA e AWS RAM
- Trasferisci z/OS dati Db2 su larga scala su Amazon S3 in file CSV

Modernizzazione

Argomenti

- Archivia automaticamente gli elementi su Amazon S3 utilizzando DynamoDB TTL
- Crea un'architettura serverless multi-tenant in Amazon Service OpenSearch
- Distribuisci applicazioni multi-stack utilizzando AWS CDK con TypeScript
- Automatizza la distribuzione di applicazioni annidate utilizzando AWS SAM
- Implementa l'isolamento dei tenant SaaS per Amazon S3 utilizzando un distributore automatico di token AWS Lambda
- Implementa il modello di saga serverless utilizzando AWS Step Functions
- Gestisci le applicazioni container locali configurando Amazon ECS Anywhere con AWS CDK
- Modernizza le applicazioni ASP.NET Web Forms su AWS
- Esegui carichi di lavoro pianificati e basati su eventi su larga scala con AWS Fargate
- Onboarding dei tenant nell'architettura SaaS per il modello a silo utilizzando C# e AWS CDK
- Scomponi i monoliti in microservizi utilizzando CQRS e l'event sourcing
- Altri modelli

Archivia automaticamente gli elementi su Amazon S3 utilizzando DynamoDB TTL

Tabby Ward, Amazon Web Services

Riepilogo

Questo modello fornisce i passaggi per rimuovere i dati più vecchi da una tabella Amazon DynamoDB e archiviarli in un bucket Amazon Simple Storage Service (Amazon S3) su Amazon Web Services (AWS) senza dover gestire una flotta di server.

Questo modello utilizza Amazon DynamoDB Time to Live (TTL) per eliminare automaticamente i vecchi elementi e Amazon DynamoDB Streams per acquisire gli elementi TTL scaduti. Quindi collega DynamoDB Streams ad AWS Lambda, che esegue il codice senza effettuare il provisioning o gestire alcun server.

Quando vengono aggiunti nuovi elementi al flusso DynamoDB, viene avviata la funzione Lambda e scrive i dati in un flusso di distribuzione Amazon Data Firehose. Firehose offre una soluzione semplice e completamente gestita per caricare i dati come archivio in Amazon S3.

DynamoDB viene spesso utilizzato per archiviare dati di serie temporali, come dati click-stream di pagine Web o dati Internet of Things (IoT) provenienti da sensori e dispositivi connessi. Invece di eliminare gli elementi a cui si accede meno frequentemente, molti clienti desiderano archiviarli per scopi di controllo. TTL semplifica questa archiviazione eliminando automaticamente gli elementi in base all'attributo timestamp.

Gli elementi eliminati tramite TTL possono essere identificati in DynamoDB Streams, che acquisisce una sequenza di modifiche a livello di elemento ordinata nel tempo e archivia la sequenza in un registro per un massimo di 24 ore. Questi dati possono essere utilizzati da una funzione Lambda e archiviati in un bucket Amazon S3 per ridurre i costi di storage. Per ridurre ulteriormente i costi, è possibile creare regole del ciclo di vita di Amazon S3 per trasferire automaticamente i dati (non appena vengono creati) alle classi di storage a basso costo.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- AWS Command Line Interface (AWS CLI) 1.7 o versione successiva, installata e configurata su macOS, Linux o Windows.

- Python 3.7 o successivo.
- Boto3, installato e configurato. Se Boto3 non è già installato, esegui il `python -m pip install boto3` comando per installarlo.

Architecture

Stack tecnologico

- Amazon DynamoDB
- Amazon DynamoDB Streams
- Amazon Data Firehose
- AWS Lambda
- Simple Storage Service (Amazon S3)

1. Gli elementi vengono eliminati tramite TTL.
2. Il trigger di flusso DynamoDB richiama la funzione del processore di flusso Lambda.
3. La funzione Lambda inserisce i record nel flusso di distribuzione di Firehose in formato batch.
4. I record di dati vengono archiviati nel bucket S3.

Tools (Strumenti)

- AWS CLI: l'AWS Command Line Interface (AWS CLI) è uno strumento unificato per gestire i servizi AWS.
- Amazon DynamoDB: Amazon DynamoDB è un database di chiave-valore e documenti che offre prestazioni a una cifra in millisecondi su qualsiasi scala.
- Amazon DynamoDB Time to Live (TTL): Amazon DynamoDB TTL ti aiuta a definire un timestamp per articolo per determinare quando un articolo non è più necessario.
- Amazon DynamoDB Streams — Amazon DynamoDB Streams acquisisce una sequenza ordinata nel tempo di modifiche a livello di elemento in qualsiasi tabella DynamoDB e archivia queste informazioni in un registro per un massimo di 24 ore.
- Amazon Data Firehose — Amazon Data Firehose è il modo più semplice per caricare in modo affidabile lo streaming di dati in data lake, data store e servizi di analisi.

- AWS Lambda: AWS Lambda esegue codice senza la necessità di effettuare il provisioning o gestire server. I costi saranno calcolati in base al tempo di elaborazione effettivo.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti che offre scalabilità, disponibilità dei dati, sicurezza e prestazioni leader del settore.

Codice

Il codice per questo modello è disponibile nel repository GitHub Archive items to S3 using DynamoDB TTL.

Epiche

Configurare una tabella DynamoDB, TTL e un flusso DynamoDB

Operazione	Description	Competenze richieste

Crea una tabella DynamoDB.	Utilizza l'AWS CLI per creare una tabella in DynamoDB chiamata. `Reservation` Scegli l'unità di capacità di lettura casuale (RCU) e l'unità di capacità di scrittura (WCU) e assegna alla tabella due attributi: e. `ReservationID` `ReservationDate`
	
	`aws dynamodb create-table \`
	`--table-name Reservation \`
	`--attribute-definitions Attribute Name=ReservationID,AttributeType=S AttributeName=ReservationDate,AttributeType=N \`
	`--key-schema Attribute Name=ReservationID`
		Architetto del cloud, sviluppatore di app

Operazione	Description	Competenze richieste

	<pre>,KeyType=HASH
	 AttributeName=Rese
	rvationDate,KeyTyp
	e=RANGE \
	--provisioned-th
	roughput ReadCapac
	ityUnits=100,Write
	CapacityUnits=100</pre>
	`ReservationDate` è un timestamp epocale che verrà utilizzato per attivare il TTL.	

Attiva DynamoDB TTL.	Utilizza l'AWS CLI per attivare DynamoDB TTL per l'attributo. `ReservationDate`
	<pre>aws dynamodb update-ti
	me-to-live \
	--table-name Reservati
	on\
	  --time-to-live-spe
	cification Enabled=t
	rue,AttributeName=
	ReservationDate</pre>
		Architetto del cloud, sviluppatore di app

Operazione	Description	Competenze richieste

Attiva uno stream DynamoDB.	Utilizza l'AWS CLI per attivare un flusso DynamoDB per la `Reservation` tabella utilizzando il tipo di flusso. `NEW_AND_OLD_IMAGES`
	
	`aws dynamodb update-table \`
	`--table-name Reservation \`
	`  --stream-specification StreamEnabled=true,StreamViewType=NEW_AND_OLD_IMAGES`
	
	Questo flusso conterrà i record relativi a nuovi elementi, elementi aggiornati, elementi eliminati e elementi eliminati tramite TTL. I record relativi agli elementi eliminati tramite TTL contengono un attributo di metadati aggiuntivo per distinguerli dagli elementi eliminati manualmente. Il `userIdentity` campo per le eliminazioni TTL indica che il servizio DynamoDB ha eseguito l'azione di eliminazione.
	
	In questo modello, vengono archiviati solo gli elementi eliminati da TTL, ma è possibile archiviare solo i
		Architetto del cloud, sviluppatore di app

Operazione	Description	Competenze richieste
	record in cui `eventName` è e contiene valori uguali a. `REMOVE` `userIdentity` `principalId dynamodb.amazonaws.com`	

Crea e configura un bucket S3

Operazione	Description	Competenze richieste
Crea un bucket S3.	Utilizza l'AWS CLI per creare un bucket S3 di destinazione nella tua regione AWS, sostituendolo `us-east-1` con la tua regione e amzn-s3-con il nome del tuo bucket. demo-destination-bucket `aws s3api create-bucket \ --bucket amzn-s3-demo-destination-bucket \ --region us-east-1` Assicurati che il nome del tuo bucket S3 sia unico a livello globale, poiché lo spazio dei nomi è condiviso da tutti gli account AWS.	Architetto del cloud, sviluppatore di app
Crea una politica del ciclo di vita di 30 giorni per il bucket S3.	1. Accedi alla Console di gestione AWS e apri la console Amazon S3. 2. Scegliete il bucket S3 che contiene i dati di Firehose.	Architetto del cloud, sviluppatore di app

Operazione	Description	Competenze richieste
	3. Nel bucket S3, scegli la scheda Gestione e scegli Aggiungi regola del ciclo di vita. 4. Inserisci un nome per la regola nella finestra di dialogo delle regole del ciclo di vita e configura una regola del ciclo di vita di 30 giorni per il tuo bucket.	

Creare un flusso di distribuzione Firehose

Operazione	Description	Competenze richieste
Crea e configura un flusso di distribuzione Firehose.	Scarica e modifica l'esempio di `CreateFireHoseToS3.py` codice dal GitHub repository. Questo codice è scritto in Python e mostra come creare un flusso di distribuzione Firehose e un ruolo AWS Identity and Access Management (IAM). Il ruolo IAM avrà una policy che può essere utilizzata da Firehose per scrivere nel bucket S3 di destinazione. Per eseguire lo script, utilizzate i seguenti argomenti di comando e riga di comando.	Architetto del cloud, sviluppatore di app

Operazione	Description	Competenze richieste
	Argomento 1=<Your_S3_bucket_ARN> , che è l'Amazon Resource Name (ARN) per il bucket creato in precedenza Argomento 2= Il nome del tuo Firehose (questo pilota lo sta firehose_to_s3_stream usando). Argomento 3= Il nome del ruolo IAM (questo programma pilota lo utilizza). firehose_to_s3 `python CreateFireHoseToS3.py <Your_S3_Bucket_ARN> firehose_to_s3_stream firehose_to_s3` Se il ruolo IAM specificato non esiste, lo script creerà un ruolo di assunzione con una politica di relazione affidabile, nonché una politica che concede autorizzazioni Amazon S3 sufficienti. Per esempi di queste politiche, consulta la sezione Informazioni aggiuntive.	

Operazione	Description	Competenze richieste
Verifica lo stream di distribuzione di Firehose.	Descrivi il flusso di distribuzione di Firehose utilizzando la CLI di AWS per verificare che il flusso di distribuzione sia stato creato correttamente. `aws firehose describe-delivery-stream --delivery-stream-name firehose_to_s3_stream`	Architetto del cloud, sviluppatore di app

Creare una funzione Lambda per elaborare il flusso di distribuzione di Firehose

Operazione	Description	Competenze richieste
Crea una politica di fiducia per la funzione Lambda.	Crea un file di criteri di fiducia con le seguenti informazioni. `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "Service": "lambda.amazonaws.com" }, "Action": "sts:AssumeRole" } ] }`	Architetto del cloud, sviluppatore di app

Operazione	Description	Competenze richieste
	Ciò consente alla funzione di accedere alle risorse AWS.	
Crea un ruolo di esecuzione per la funzione Lambda.	Per creare il ruolo di esecuzione, esegui il codice seguente. `aws iam create-role --role-name lambda-ex --assume-role-policy-document file://TrustPolicy.json`	Architetto del cloud, sviluppatore di app

Operazione	Description	Competenze richieste

Aggiungi l'autorizzazione al ruolo.	Per aggiungere l'autorizzazione al ruolo, usa il `attach-policy-to-role` comando.
	`aws iam attach-role-policy --role-name lambda-ex --policy-arn arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole`
	`aws iam attach-role-policy --role-name lambda-ex --policy-arn arn:aws:iam::aws:policy/service-role/AWSLambdaDynamoDBExecutionRole`
	`aws iam attach-role-policy --role-name lambda-ex --policy-arn arn:aws:iam::aws:policy/AmazonKinesisFirehoseFullAccess`
	`aws iam attach-role-policy --role-name lambda-ex --policy-arn arn:aws:iam::aws:policy/IAMFullAccess`
		Architetto del cloud, sviluppatore di app

Operazione	Description	Competenze richieste

Creazione di una funzione Lambda.	Comprimi il `LambdaStreamProcessor.py` file dal repository di codice eseguendo il seguente comando.
	`zip function.zip LambdaStreamProcessor.py`
	Quando crei la funzione Lambda, avrai bisogno del ruolo di esecuzione Lambda ARN. Per ottenere l'ARN, esegui il codice seguente.
	`aws iam get-role \ --role-name lambda-ex`
	Per creare la funzione Lambda, esegui il codice seguente.
	`# Review the environment variables and replace them with your values.`
	`aws lambda create-function --function-name LambdaStreamProcessor \ --zip-file fileb://function.zip --handler LambdaStreamProcessor.handler --runtime python3.8 \`
		Architetto del cloud, sviluppatore di app

Operazione	Description	Competenze richieste
	`--role {Your Lamda Execution Role ARN}\ --environment Variables="{firehose_name=firehose_to_s3_stream,bucket_arn = <Your_S3_bucket_ARN>,iam_role_name = firehose_to_s3, batch_size=400}"`	
Configura il trigger della funzione Lambda.	Utilizza l'AWS CLI per configurare il trigger (DynamoDB Streams), che richiama la funzione Lambda. La dimensione del batch di 400 serve per evitare di incorrere in problemi di concorrenza Lambda. `aws lambda create-event-source-mapping --function-name LambdaStreamProcessor \ --batch-size 400 --starting-position LATEST \ --event-source-arn <Your Latest Stream ARN From DynamoDB Console>`	Architetto del cloud, sviluppatore di app

Prova la funzionalità

Operazione	Description	Competenze richieste
Aggiungi articoli con timestamp scaduti alla tabella delle prenotazioni.	Per testare la funzionalità, aggiungi alla tabella elementi con timestamp d'epoca scaduti. `Reservation TTL` eliminerà automaticamente gli elementi in base al timestamp . La funzione Lambda viene avviata sulle attività di DynamoDB Stream e filtra l'evento per identificare l'attività o gli elementi eliminati. `REMOVE` Quindi inserisce i record nel flusso di distribuzione di Firehose in formato batch. Il flusso di distribuzione Firehose trasferisce gli articoli a un bucket S3 di destinazione con il prefisso. `firehosetos3example/year=current year/month=current month/day=current day/hour=current hour/` **Important** Per ottimizzare il recupero dei dati, configura Amazon S3 con	Architetto del cloud

Operazione	Description	Competenze richieste
	`Prefix` `ErrorOutputPrefix` le informazioni dettagliate nella sezione Informazioni aggiuntive.	

Pulisci le risorse

Operazione	Description	Competenze richieste
Eliminare tutte le risorse.	Elimina tutte le risorse per assicurarti che non ti vengano addebitati costi per i servizi che non utilizzi.	Architetto del cloud, sviluppatore di app

Risorse correlate

- Gestione del ciclo di vita dello storage
- Classi di storage Amazon S3
- Documentazione dell'SDK AWS per Python (Boto3)

Informazioni aggiuntive

Creare e configurare un flusso di distribuzione Firehose — Esempi di policy

Documento di esempio sulla politica delle relazioni di fiducia di Firehose

```
firehose_assume_role = {
        'Version': '2012-10-17',
        'Statement': [
            {
                'Sid': '',
                'Effect': 'Allow',
```

```
                'Principal': {
                    'Service': 'firehose.amazonaws.com'
                },
                'Action': 'sts:AssumeRole'
            }
        ]
    }
```

Esempio di politica di autorizzazione S3

```
s3_access = {
        "Version": "2012-10-17",
        "Statement": [
            {
                "Sid": "",
                "Effect": "Allow",
                "Action": [
                    "s3:AbortMultipartUpload",
                    "s3:GetBucketLocation",
                    "s3:GetObject",
                    "s3:ListBucket",
                    "s3:ListBucketMultipartUploads",
                    "s3:PutObject"
                ],
                "Resource": [
                    "{your s3_bucket ARN}/*",
                    "{Your s3 bucket ARN}"
                ]
            }
        ]
    }
```

Verifica la funzionalità: configurazione Amazon S3

La configurazione Amazon S3 con la seguente `Prefix` e `ErrorOutputPrefix` viene scelta per ottimizzare il recupero dei dati.

prefisso

```
firehosetos3example/year=! {timestamp: yyyy}/month=! {timestamp:MM}/day=!
 {timestamp:dd}/hour=!{timestamp:HH}/
```

Firehose crea innanzitutto una cartella di base chiamata `firehosetos3example` direttamente sotto il bucket S3. Quindi valuta le espressioni e l'anno `!{timestamp:yyyy}!{timestamp:MM}`, il mese`!{timestamp:dd}`, `!{timestamp:HH}` il giorno e l'ora utilizzando il formato Java. DateTimeFormatter

Ad esempio, un timestamp di arrivo approssimativo di 1604683577 in Unix epoch Time restituisce,, e. `year=2020` `month=11` `day=06` `hour=05` Pertanto, viene valutata la posizione in Amazon S3, in cui vengono distribuiti i record di dati. `firehosetos3example/year=2020/month=11/day=06/hour=05/`

ErrorOutputPrefix

```
firehosetos3erroroutputbase/!{firehose:random-string}/!{firehose:error-output-type}/!{timestamp:yyyy/MM/dd}/
```

I `ErrorOutputPrefix` risultati sono in una cartella di base richiamata `firehosetos3erroroutputbase` direttamente sotto il bucket S3. L'espressione `!{firehose:random-string}` restituisce una stringa casuale di 11 caratteri come. `ztWxkdg3Thg` Potrebbe essere valutata la posizione di un oggetto Amazon S3 in cui vengono consegnati i record non riusciti. `firehosetos3erroroutputbase/ztWxkdg3Thg/processing-failed/2020/11/06/`

Crea un'architettura serverless multi-tenant in Amazon Service OpenSearch

Tabby Ward e Nisha Gambhir, Amazon Web Services

Riepilogo

Amazon OpenSearch Service è un servizio gestito che semplifica l'implementazione, il funzionamento e la scalabilità di Elasticsearch, un popolare motore di ricerca e analisi open source. OpenSearch Il servizio offre la ricerca a testo libero, nonché l'inserimento e la creazione di dashboard quasi in tempo reale per lo streaming di dati come log e metriche.

I fornitori di software as a service (SaaS) utilizzano spesso OpenSearch Service per affrontare un'ampia gamma di casi d'uso, ad esempio per ottenere informazioni sui clienti in modo scalabile e sicuro, riducendo al contempo la complessità e i tempi di inattività.

L'utilizzo OpenSearch del servizio in un ambiente multi-tenant introduce una serie di considerazioni che influiscono sul partizionamento, l'isolamento, l'implementazione e la gestione della soluzione SaaS. I provider SaaS devono considerare come scalare efficacemente i propri cluster Elasticsearch con carichi di lavoro in continuo cambiamento. Devono inoltre considerare in che modo la suddivisione in più livelli e le condizioni rumorose dei vicini potrebbero influire sul loro modello di partizionamento.

Questo modello esamina i modelli utilizzati per rappresentare e isolare i dati dei tenant con costrutti Elasticsearch. Inoltre, il modello si concentra su una semplice architettura di riferimento serverless come esempio per dimostrare l'indicizzazione e la ricerca utilizzando Service in un ambiente multi-tenant. OpenSearch Implementa il modello di partizionamento dei dati del pool, che condivide lo stesso indice tra tutti i tenant mantenendo l'isolamento dei dati del tenant. Questo modello utilizza i seguenti AWS servizi: Amazon API Gateway AWS Lambda, Amazon Simple Storage Service (Amazon S3) e Service. OpenSearch

Per ulteriori informazioni sul modello di pool e su altri modelli di partizionamento dei dati, consulta la sezione Informazioni aggiuntive.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- AWS Command Line Interface (AWS CLI) versione 2.x, installata e configurata su macOS, Linux o Windows

- Python versione 3.9
- pip3 — Il codice sorgente di Python viene fornito come file.zip da distribuire in una funzione Lambda. Se desideri utilizzare il codice localmente o personalizzarlo, segui questi passaggi per sviluppare e ricompilare il codice sorgente:
  1. Genera il `requirements.txt` file eseguendo il seguente comando nella stessa directory degli script Python: `pip3 freeze > requirements.txt`
  2. Installa le dipendenze: `pip3 install -r requirements.txt`

Limitazioni

- Questo codice viene eseguito in Python e attualmente non supporta altri linguaggi di programmazione.
- L'applicazione di esempio non include il supporto AWS interregionale o di disaster recovery (DR).
- Questo modello è destinato esclusivamente a scopo dimostrativo. Non è destinato all'uso in un ambiente di produzione.

Architecture

Il diagramma seguente illustra l'architettura di alto livello di questo pattern. L'architettura include quanto segue:

- Lambda per indicizzare e interrogare il contenuto
- OpenSearch Servizio per eseguire ricerche
- API Gateway per fornire un'interazione API con l'utente
- Amazon S3 per archiviare dati grezzi (non indicizzati)
- Amazon CloudWatch per monitorare i log
- AWS Identity and Access Management (IAM) per creare ruoli e politiche degli inquilini

Automazione e scalabilità

Per semplicità, il pattern utilizza AWS CLI il provisioning dell'infrastruttura e la distribuzione del codice di esempio. È possibile creare uno CloudFormation o più AWS Cloud Development Kit (AWS CDK) script per automatizzare il pattern.

Tools (Strumenti)

Servizi AWS

- AWS CLIè uno strumento unificato per la gestione Servizi AWS e le risorse utilizzando i comandi nella shell della riga di comando.
- Lambda è un servizio di elaborazione che consente di eseguire codice senza effettuare il provisioning o la gestione di server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo.
- API Gateway Servizio AWS consente di creare, pubblicare, mantenere, monitorare e proteggere REST, HTTP e WebSocket APIs su qualsiasi scala.
- Amazon S3 è un servizio di storage di oggetti che consente di archiviare e recuperare qualsiasi quantità di informazioni in qualsiasi momento, da qualsiasi punto del Web.
- OpenSearch Service è un servizio completamente gestito che semplifica l'implementazione, la protezione e l'esecuzione di Elasticsearch su larga scala in modo conveniente.

Codice

L'allegato fornisce file di esempio per questo modello. Ciò include:

- `index_lambda_package.zip`— La funzione Lambda per l'indicizzazione dei dati in OpenSearch Service utilizzando il modello pool.
- `search_lambda_package.zip`— La funzione Lambda per la ricerca di dati in OpenSearch Service.
- `Tenant-1-data`— Esempio di dati grezzi (non indicizzati) per Tenant-1.
- `Tenant-2-data`— Esempio di dati grezzi (non indicizzati) per Tenant-2.

> **Important**
>
> Le storie di questo modello includono esempi di AWS CLI comandi formattati per Unix, Linux e macOS. Per Windows, sostituisci il carattere di continuazione UNIX barra rovesciata (\) al termine di ogni riga con un accento circonflesso (^).

> **Note**
>
> Nei AWS CLI comandi, sostituite tutti i valori all'interno delle parentesi angolari (<>) con i valori corretti.

Epiche

Crea e configura un bucket S3

Operazione	Description	Competenze richieste

Crea un bucket S3.	Crea un bucket S3 nel tuo. Regione AWS Questo bucket conterrà i dati del tenant non indicizzati per l'applicazione di esempio. Assicurati che il nome del bucket S3 sia univoco a livello globale, perché lo spazio dei nomi è condiviso da tutti. Account AWS
	
	Per creare un bucket S3, puoi usare il comando create-bucket come segue: AWS CLI
	
	`aws s3api create-bucket \`
	`  --bucket <tenantrawdata> \`
	`  --region <your-AWS-Region>`
	
	dov'è il nome del bucket `tenantrawdata` S3. (È possibile utilizzare qualsiasi nome univoco che segua le
		Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste
	linee guida per la denominazione dei bucket.)	

Crea e configura un cluster Elasticsearch

Operazione	Description	Competenze richieste
Crea un dominio OpenSearch di servizio.	Esegui il AWS CLI create-elasticsearch-domaincomando per creare un dominio OpenSearch di servizio: `aws es create-elasticsearch-domain \ --domain-name vpc-cli-example \ --elasticsearch-version 7.10 \ --elasticsearch-cluster-config InstanceType=t3.medium.elasticsearch,InstanceCount=1 \ --ebs-options EBSEnabled=true,VolumeType=gp2,VolumeSize=10 \ --domain-endpoint-options "{\"EnforceHTTPS\": true}" \ --encryption-at-rest-options "{\"Enabled\": true}" \ --node-to-node-encryption-options "{\"Enabled\": true}" \ --advanced-security-options "{\"Enabl`	Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste
	ed\": true, \"Interna lUserDatabaseEnabled \": true, \ \"MasterUserOption s\": {\"MasterUserName \": \"KibanaUser\", \ \"MasterUserPasswo rd\": \"NewKiba naPassword@123\"}}" \ --vpc-options "{\"SubnetIds\": [\"<subnet-id>\"], \"SecurityGroupIds\": [\"<sg-id>\"]}" \ --access-policies "{\"Version\": \"2012-10-17\", \"Statement\": [{ \"Effect\": \"Allow\", \ \"Principal\": {\"AWS\": \"*\" }, \"Action\":\"es:*\", \ \"Resource\": \"arn:aws:es:<regi on>:<account-id>:d omain\/vpc-cli-exa mple\/*\" }] }" Il conteggio delle istanze è impostato su 1 perché il dominio è a scopo di test. È necessario abilitare il controllo granulare degli accessi utilizzando il `advanced-security-options` parametro, poiché i dettagli non possono essere modificat	

Operazione	Description	Competenze richieste
	i dopo la creazione del dominio. Questo comando crea un nome utente principal e (`KibanaUser`) e una password che puoi usare per accedere alla console Kibana. Poiché il dominio fa parte di un cloud privato virtuale (VPC), devi assicurarti di poter raggiungere l'istanza Elasticse arch specificando la politica di accesso da utilizzare. Per ulteriori informazioni, consulta la sezione Avvio dei domini Amazon OpenSearch Service all'interno di un VPC nella documentazione. AWS	

Operazione	Description	Competenze richieste

Configura un bastion host.	Configura un'istanza Windows di Amazon Elastic Compute Cloud (Amazon EC2) come host bastion per accedere alla console Kibana. Il gruppo di sicurezza Elasticsearch deve consentire il traffico proveniente dal gruppo di EC2 sicurezza Amazon. Per istruzioni, consulta il post sul blog Controllare l'accesso alla rete alle EC2 istanze utilizzando un server Bastion.
	
	Quando il bastion host è stato configurato e hai a disposizione il gruppo di sicurezza associato all'istanza, usa il AWS CLI authorize-security-group-ingresscomando per aggiungere l'autorizzazione al gruppo di sicurezza Elasticsearch per consentire la porta 443 dal gruppo di sicurezza Amazon EC2 (bastion host).
	
	`aws ec2 authorize-security-group-ingress \`
	`  --group-id <SecurityGroupIdfElasticSearch> \`
	`  --protocol tcp \`
	`  --port 443 \`
		Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste
	`--source-group <SecurityGroupIdfBashionHostEC2>`	

Creare e configurare la funzione di indice Lambda

Operazione	Description	Competenze richieste

Crea il ruolo di esecuzione Lambda.	Esegui il comando AWS CLI create-role per concedere alla funzione di indice Lambda l'accesso a e risorse: Servizi AWS
	`aws iam create-role \ --role-name index-lambda-role \ --assume-role-policy-document file://lambda_assume_role.json`
	dove `lambda_assume_role.json` è un documento JSON che concede `AssumeRole` le autorizzazioni alla funzione Lambda, come segue:
	`{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": {`
		Architetto cloud, amministratore cloud

Operazione	Description	Competenze richieste
	"Service": "lambda.amazonaws.com" }, "Action": "sts:AssumeRole" }] }	

Operazione	Description	Competenze richieste

Associa policy gestite al ruolo Lambda.	Esegui il AWS CLI attach-role-policycomando per allegare le politiche gestite al ruolo creato nel passaggio precedente. Queste due politiche forniscono al ruolo le autorizzazioni per creare un'interfaccia di rete elastica e scrivere log su Logs. CloudWatch
	
	`aws iam attach-role-policy \ --role-name index-lambda-role \ --policy-arn arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole`
	
	`aws iam attach-role-policy \ --role-name index-lambda-role \ --policy-arn arn:aws:iam::aws:policy/service-role/AWSLambdaVPCAccessExecutionRole`
		Architetto cloud, amministratore cloud

Operazione	Description	Competenze richieste

Crea una politica per concedere alla funzione di indice Lambda il permesso di leggere gli oggetti S3.	Esegui il comando AWS CLI create-policy per concedere alla funzione di indice Lambda `s3:GetObject` il permesso di leggere gli oggetti nel bucket S3:
	`aws iam create-policy \ --policy-name s3-permission-policy \ --policy-document file://s3-policy.json`
	Il file `s3-policy.json` è un documento JSON mostrato di seguito che concede le `s3:GetObject` autorizzazioni per consentire l'accesso in lettura agli oggetti S3. Se hai usato un nome diverso quando hai creato il bucket S3, fornisci il nome del bucket corretto nella sezione: `Resource`
	`{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": "s3:GetObject", "Resource": "arn:aws:s3:::<tenantrawdata>/*"`
		Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste
	<pre> }] }</pre>	

Allega la politica di autorizzazione di Amazon S3 al ruolo di esecuzione Lambda.	Esegui il AWS CLI attach-role-policycomando per allegare la politica di autorizzazione di Amazon S3 creata nel passaggio precedente al ruolo di esecuzione Lambda:
	<pre>aws iam attach-role-policy \
	  --role-name index-lambda-role \
	  --policy-arn <PolicyARN></pre>
	`PolicyARN` dov'è l'Amazon Resource Name (ARN) della politica di autorizzazione di Amazon S3. Puoi ottenere questo valore dall'output del comando precedente.
		Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste

Crea la funzione di indice Lambda.	Esegui il comando AWS CLI create-function per creare la funzione di indice Lambda, che accederà a Service: OpenSearch
	
	`aws lambda create-function \`
	`  --function-name index-lambda-function \`
	`  --zip-file fileb://index_lambda_package.zip \`
	`  --handler lambda_index.lambda_handler \`
	`  --runtime python3.9 \`
	`  --role "arn:aws:iam::account-id:role/index-lambda-role" \`
	`  --timeout 30 \`
	`  --vpc-config "{\"SubnetIds\": [\"<subnet-id1\>", \"<subnet-id2>\"], \`
	`  \"SecurityGroupIds\": [\"<sg-1>\"]}"`
		Architetto cloud, amministratore cloud

Operazione	Description	Competenze richieste
Consenti ad Amazon S3 di chiamare la funzione di indice Lambda.	Esegui il comando AWS CLI add-permission per concedere ad Amazon S3 l'autorizzazione a chiamare la funzione di indice Lambda: `aws lambda add-permission \ --function-name index-lambda-function \ --statement-id s3-permissions \ --action lambda:InvokeFunction \ --principal s3.amazonaws.com \ --source-arn "arn:aws:s3:::<tenantrawdata>" \ --source-account "<account-id>"`	Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste

Aggiungi un trigger Lambda per l'evento Amazon S3.	Esegui il AWS CLI put-bucket-notification-configuration comando per inviare notifiche alla funzione di indice Lambda quando viene rilevato l'evento Amazon `ObjectCreated` S3. La funzione index viene eseguita ogni volta che un oggetto viene caricato nel bucket S3.
	
	`aws s3api put-bucket-notification-configuration \`
	`  --bucket <tenantrawdata> \`
	`  --notification-configuration file://s3-trigger.json`
	
	Il file `s3-trigger.json` è un documento JSON nella cartella corrente che aggiunge la policy delle risorse alla funzione Lambda quando si verifica l'evento Amazon `ObjectCreated` S3.
		Architetto del cloud, amministratore del cloud

Creare e configurare la funzione di ricerca Lambda

Operazione	Description	Competenze richieste
Crea il ruolo di esecuzione Lambda.	Esegui il comando AWS CLI create-role per concedere alla funzione di ricerca Lambda	Architetto cloud, amministratore cloud

Operazione	Description	Competenze richieste

	l'accesso a e risorse: Servizi AWS
	
	`aws iam create-role \ --role-name search-lambda-role \ --assume-role-policy-document file://lambda_assume_role.json`
	
	dove `lambda_assume_role.json` è un documento JSON nella cartella corrente che concede `AssumeRole` le autorizzazioni alla funzione Lambda, come segue:
	
	`{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "Service": "lambda.amazonaws.com" }, "Action": "sts:AssumeRole" } ] }`	

Operazione	Description	Competenze richieste

Associa policy gestite al ruolo Lambda.	Esegui il AWS CLI attach-role-policycomando per allegare le politiche gestite al ruolo creato nel passaggio precedente. Queste due politiche forniscono al ruolo le autorizzazioni per creare un'interfaccia di rete elastica e scrivere log su Logs. CloudWatch
	
	`aws iam attach-role-policy \ --role-name search-lambda-role \ --policy-arn arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole`
	
	`aws iam attach-role-policy \ --role-name search-lambda-role \ --policy-arn arn:aws:iam::aws:policy/service-role/AWSLambdaVPCAccessExecutionRole`
		Architetto cloud, amministratore cloud

Operazione	Description	Competenze richieste
Crea la funzione di ricerca Lambda.	Esegui il comando AWS CLI create-function per creare la funzione di ricerca Lambda, che accederà a Service: OpenSearch `aws lambda create-function \ --function-name search-lambda-function \ --zip-file fileb://search_lambda_package.zip \ --handler lambda_search.lambda_handler \ --runtime python3.9 \ --role "arn:aws:iam::account-id:role/search-lambda-role" \ --timeout 30 \ --vpc-config "{\"SubnetIds\": [\"<subnet-id1\>", \"<subnet-id2>\"], \ \"SecurityGroupIds\": [\"<sg-1>\"]}"`	Architetto cloud, amministratore cloud

Crea e configura i ruoli degli inquilini

Operazione	Description	Competenze richieste
Crea ruoli IAM tenant.	Esegui il comando AWS CLI create-role per creare due ruoli tenant che verranno utilizzati per testare la funzionalità di ricerca:	Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste
	`aws iam create-role \ --role-name Tenant-1-role \ --assume-role-policy-document file://assume-role-policy.json`	
	`aws iam create-role \ --role-name Tenant-2-role \ --assume-role-policy-document file://assume-role-policy.json`	
	Il file `assume-role-policy.json` è un documento JSON nella cartella corrente che concede le `AssumeRole` autorizzazioni per il ruolo di esecuzione Lambda:	
	`{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "AWS": "<Lambda execution role for index function>", "AWS": "<Lambda execution role for search function>" }, `	

Operazione	Description	Competenze richieste

	<pre>"Action":
	"sts:AssumeRole"
	}
	]
	}</pre>	

Operazione	Description	Competenze richieste

Crea una policy IAM per i tenant.	Esegui il comando AWS CLI create-policy per creare una politica del tenant che garantisca l'accesso alle operazioni di Elasticsearch:
	`aws iam create-policy \ --policy-name tenant-policy \ --policy-document file://policy.json`
	Il file `policy.json` è un documento JSON nella cartella corrente che concede le autorizzazioni su Elasticsearch:
	`{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [ "es:ESHttpDelete", "es:ESHttpGet", "es:ESHttpHead", "es:ESHttpPost", "es:ESHttpPut", "es:ESHttpPatch" ],`
		Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste
	<pre>"Resource": ["<ARN of Elasticsearch domain created earlier>"] }] }</pre>	

Allega la policy IAM del tenant ai ruoli del tenant.	Esegui il AWS CLI attach-role-policycomando per allegare la politica IAM del tenant ai due ruoli tenant che hai creato nel passaggio precedente:
	<pre>aws iam attach-role-policy \ --policy-arn arn:aws:iam::account-id:policy/tenant-policy \ --role-name Tenant-1-role
	
	aws iam attach-role-policy \ --policy-arn arn:aws:iam::account-id:policy/tenant-policy \ --role-name Tenant-2-role</pre>
	L'ARN della policy proviene dall'output del passaggio precedente.
		Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste

Crea una policy IAM per concedere a Lambda le autorizzazioni per assumere il ruolo.	Esegui il comando AWS CLI create-policy per creare una politica affinché Lambda assuma il ruolo di tenant:
	`aws iam create-policy \ --policy-name assume-tenant-role-policy \ --policy-document file://lambda_policy.json`
	Il file `lambda_policy.json` è un documento JSON nella cartella corrente che concede le autorizzazioni per: `AssumeRole`
	`{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": "sts:AssumeRole", "Resource": "<ARN of tenant role created earlier>" } ] }`
	InfattiResource, puoi usare un carattere jolly per evitare di
		Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste
	creare una nuova politica per ogni tenant.	

Crea una policy IAM per concedere al ruolo dell'indice Lambda l'autorizzazione ad accedere ad Amazon S3.	Esegui il comando AWS CLI create-policy per concedere al ruolo dell'indice Lambda il permesso di accedere agli oggetti nel bucket S3:
	`aws iam create-policy \ --policy-name s3-permission-policy \ --policy-document file://s3_lambda_policy.json`
	Il file `s3_lambda_policy.json` è il seguente documento di policy JSON nella cartella corrente:
	`{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": "s3:GetObject", "Resource": "arn:aws:s3:::tenantrawdata/*" } ] }`
		Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste

Associa la policy al ruolo di esecuzione Lambda.	Esegui il AWS CLI attach-role-policycomando per allegare la policy creata nel passaggio precedente all'indice Lambda e ai ruoli di esecuzione della ricerca che hai creato in precedenza:
	
	`aws iam attach-role-policy \`
	`  --policy-arn arn:aws:iam::account-id:policy/assume-tenant-role-policy \`
	`  --role-name index-lambda-role`
	
	`aws iam attach-role-policy \`
	`  --policy-arn arn:aws:iam::account-id:policy/assume-tenant-role-policy \`
	`  --role-name search-lambda-role`
	
	`aws iam attach-role-policy \`
	`  --policy-arn arn:aws:iam::account-id:policy/s3-permission-policy \`
	`  --role-name index-lambda-role`
	
	L'ARN della policy proviene dall'output del passaggio precedente.
		Architetto del cloud, amministratore del cloud

Crea e configura un'API di ricerca

Operazione	Description	Competenze richieste
Crea un'API REST in API Gateway.	Esegui il AWS CLI create-rest-apicomando per creare una risorsa API REST: `aws apigateway create-rest-api \ --name Test-Api \ --endpoint-configuration "{ \"types\": [\"REGIONAL\"] }"` Per il tipo di configurazione dell'endpoint, è possibile specificare `EDGE` anziché `REGIONAL` utilizzare posizioni periferiche anziché una particolare Regione AWS. Annotate il valore del `id` campo dall'output del comando. Questo è l'ID API che utilizzerai nei comandi successivi.	Architetto del cloud, amministratore del cloud
Crea una risorsa per l'API di ricerca.	La risorsa API di ricerca avvia la funzione di ricerca Lambda con il nome della risorsa. `search` (Non è necessario creare un'API per la funzione di indice Lambda, perché viene eseguita automaticamente quando gli oggetti vengono caricati nel bucket S3.)	Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste

	1. Esegui il comando AWS CLI get-resources per ottenere l'ID principale per il percorso principale:
	`aws apigateway get-resources \ --rest-api-id <API-ID>`
	Nota il valore del campo ID. Utilizzerai questo ID principale nel comando successivo.
	`{ "items": [ { "id": "zpsri964ck", "path": "/" } ] }`
	2. Esegui il comando AWS CLI create-resource per creare una risorsa per l'API di ricerca. Per `parent-id`, specifica l'ID del comando precedente.
	`aws apigateway create-resource \ --rest-api-id <API-ID> \`	

Operazione	Description	Competenze richieste
	`--parent-id <Parent-ID> \ --path-part search`	
Crea un metodo GET per l'API di ricerca.	Esegui il comando AWS CLI put-method per creare un `GET` metodo per l'API di ricerca: `aws apigateway put-method \ --rest-api-id <API-ID> \ --resource-id <ID from the previous command output> \ --http-method GET \ --authorization-type "NONE" \ --no-api-key-required` Per `resource-id` , specifica l'ID dall'output del `create-resource` comando.	Architetto cloud, amministratore cloud

Operazione	Description	Competenze richieste
Crea un metodo di risposta per l'API di ricerca.	Esegui il AWS CLI put-method-responsecomando per aggiungere un metodo di risposta per l'API di ricerca: `aws apigateway put-method-response \ --rest-api-id <API-ID> \ --resource-id <ID from the create-resource command output> \ --http-method GET \ --status-code 200 \ --response-models "{\"application/json\": \"Empty\"}"` Per `resource-id` , specifica l'ID dall'output del `create-resource` comando precedente.	Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste
Configura un'integrazione proxy Lambda per l'API di ricerca.	Esegui il comando AWS CLI put-integration per impostare un'integrazione con la funzione di ricerca Lambda: `aws apigateway put-integration \ --rest-api-id <API-ID> \ --resource-id <ID from the create-resource command output> \ --http-method GET \ --type AWS_PROXY \ --integration-http-method GET \ --uri arn:aws:apigateway:region:lambda:path/2015-03-31/functions/arn:aws:lambda:<region>:<account-id>:function:<function-name>/invocations`  Per `resource-id`, specifica l'ID del comando precedente. `create-resource`	Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste
Concedi l'autorizzazione API Gateway per chiamare la funzione di ricerca Lambda.	Esegui il comando AWS CLI add-permission per autorizzare API Gateway a utilizzare la funzione di ricerca: `aws lambda add-permission \ --function-name <function-name> \ --statement-id apigateway-get \ --action lambda:InvokeFunction \ --principal apigateway.amazonaws.com \ --source-arn "arn:aws:execute-api:<region>:<account-id>:api-id/*/GET/search"` Modifica il `source-arn` percorso se hai utilizzato un nome di risorsa API diverso anzichésearch.	Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste

Implementa l'API di ricerca.	Esegui il comando AWS CLI create-deployment per creare una risorsa di fase denominata: `dev`
	
	`aws apigateway create-deployment \`
	`  --rest-api-id <API-ID> \`
	`  --stage-name dev`
	
	Se aggiorni l'API, puoi utilizzare lo stesso AWS CLI comando per ridistribuirla nella stessa fase.
		Architetto del cloud, amministratore del cloud

Crea e configura i ruoli di Kibana

Operazione	Description	Competenze richieste
Accedi alla console Kibana.	1. Trova il link a Kibana nella dashboard del tuo dominio nella console di servizio. OpenSearch L'URL è nel formato:. `<domain-endpoint>/_plugin/kibana/` 2. Usa il bastion host che hai configurato nella prima epic per accedere alla console Kibana. 3. Accedi alla console Kibana utilizzando il nome utente principale e la password	Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste
	del passaggio precedente, quando hai creato il dominio del servizio. OpenSearch 4. Quando viene richiesto di selezionare un inquilino, scegli Privato.	

Operazione	Description	Competenze richieste
Crea e configura i ruoli di Kibana.	Per garantire l'isolamento dei dati e assicurarsi che un tenant non possa recuperare i dati di un altro tenant, è necessario utilizzare la sicurezza dei documenti, che consente agli inquilini di accedere solo ai documenti che contengono il loro ID tenant. 1. Sulla console Kibana, nel pannello di navigazione, scegli Sicurezza, Ruolo. 2. Crea un nuovo ruolo di tenant. 3. Imposta le autorizzazioni del cluster su `indices_all` , che fornisce le autorizzazioni di creazione , lettura, aggiornamento ed eliminazione (CRUD) sull'indice del servizio. OpenSearch 4. Limita le autorizzazioni dell'indice all'indice. `tenant-data` (Il nome dell'indice deve corrispondere al nome nelle funzioni di ricerca e indice Lambda.) 5. Imposta i permessi di indicizzazione su `indices_all` , per	Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste
	consentire agli utenti di eseguire tutte le operazioni relative all'indice. (Puoi limitare le operazioni per un accesso più granulare, a seconda delle tue esigenze.) 6. Per la sicurezza a livello di documento, utilizza la seguente politica per filtrare i documenti in base all'ID del tenant, per fornire l'isolamento dei dati ai tenant in un indice condiviso: `{ "bool": { "must": { "match": { "TenantId": "Tenant-1" } } } }` Il nome, le proprietà e i valori dell'indice fanno distinzione tra maiuscole e minuscole.	

Operazione	Description	Competenze richieste
Associa gli utenti ai ruoli.	1. Scegli la scheda Utenti mappati per il ruolo, quindi scegli Mappa utenti. 2. Nella sezione Ruoli di backend, specifica l'ARN del ruolo tenant IAM creato in precedenza, quindi scegli Mappa. Questo associa il ruolo del tenant IAM al ruolo Kibana in modo che la ricerca specifica del tenant restituisca i dati solo per quel tenant. Ad esempio, se il nome del ruolo IAM per Tenant-1 è`Tenant-1-Role` , specifica l'ARN per `Tenant-1-Role` (dall'epopea Crea e configura i ruoli tenant) nella casella Ruoli di backend per il ruolo Tenant-1 Kibana. 3. Ripetere i passaggi 1 e 2 per Tenant-2. Ti consigliamo di automatizzare la creazione dei ruoli tenant e Kibana al momento dell'onboarding del tenant.	Architetto cloud, amministratore cloud

Operazione	Description	Competenze richieste
Crea l'indice dei dati degli inquilini.	Nel riquadro di navigazione, in Gestione, scegli Dev Tools, quindi esegui il comando seguente. Questo comando crea l'`tenant-data` indice per definire la mappatura della `TenantId` proprietà. `PUT /tenant-data { "mappings": { "properties": { "TenantId": { "type": "keyword"} } } }`	Architetto del cloud, amministratore del cloud

Crea endpoint VPC per Amazon S3 e AWS STS

Operazione	Description	Competenze richieste
Crea un endpoint VPC per Amazon S3.	Esegui il AWS CLI create-vpc-endpointcomando per creare un endpoint VPC per Amazon S3. L'endpoint abilita la funzione di indice Lambda nel VPC per accedere ad Amazon S3. `aws ec2 create-vpc-endpoint \ --vpc-id <VPC-ID> \ --service-name com.amazonaws.us-east-1.s3 \`	Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste
	`--route-table-ids <route-table-ID>` Per `vpc-id`, specifica il VPC che stai utilizzando per la funzione di indice Lambda. Per `service-name`, usa l'URL corretto per l'endpoint Amazon S3. Per `route-table-ids`, specifica la tabella di routing associata all'endpoint VPC.	

Operazione	Description	Competenze richieste
Crea un endpoint VPC per. AWS STS	Esegui il AWS CLI create-vpc-endpointcomando per creare un endpoint VPC per AWS Security Token Service ().AWS STS L'endpoint consente l'accesso all'indice Lambda e alle funzioni di ricerca nel VPC. AWS STS Le funzioni AWS STS vengono utilizzate quando assumono il ruolo IAM. `aws ec2 create-vpc-endpoint \ --vpc-id <VPC-ID> \ --vpc-endpoint-type Interface \ --service-name com.amazonaws.us-east-1.sts \ --subnet-id <subnet-ID> \ --security-group-id <security-group-ID>` Per`vpc-id`, specifica il VPC che stai utilizzando per l'indice Lambda e le funzioni di ricerca. Infatti`subnet-id` , fornisci la sottorete in cui deve essere creato questo endpoint. Per`security-group-id` , specifica il gruppo di sicurezza a cui associare questo endpoint. (Potrebbe essere lo stesso del	Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste
	gruppo di sicurezza utilizzato da Lambda).	

Testa la multi-tenancy e l'isolamento dei dati

Operazione	Description	Competenze richieste
Aggiorna i file Python per le funzioni di indice e ricerca.	1. Nel `index_lambda_package.zip` file, modifica il `lamba_index.py` file per aggiornare l' Account AWS ID e le informazioni sull' Regione AWS endpoint Elasticsearch. 2. Nel `search_lambda_package.zip` file, modifica il `lambda_search.py` file per aggiornare l' Account AWS ID e le informazioni sull'endpoint Elasticsearch. Regione AWS Puoi ottenere l'endpoint Elasticsearch dalla scheda Panoramica della console di servizio. OpenSearch Ha il formato. `<AWS-Region>.es.amazonaws.com`	Architetto del cloud, sviluppatore di app
Aggiorna il codice Lambda.	Usa il AWS CLI update-function-codecomando per	Architetto cloud, sviluppatore di app

Operazione	Description	Competenze richieste

	aggiornare il codice Lambda con le modifiche apportate ai file Python:
	
	`aws lambda update-function-code \`
	`  --function-name index-lambda-function \`
	`  --zip-file fileb://index_lambda_package.zip`
	
	`aws lambda update-function-code \`
	`  --function-name search-lambda-function \`
	`  --zip-file fileb://search_lambda_package.zip`	

Operazione	Description	Competenze richieste

Carica i dati grezzi nel bucket S3.	Usa il comando AWS CLI cp per caricare i dati per gli oggetti Tenant-1 e Tenant-2 nel `tenantrawdata` bucket (specifica il nome del bucket S3 che hai creato a questo scopo):
	`aws s3 cp tenant-1-data s3://tenantrawdata`
	`aws s3 cp tenant-2-data s3://tenantrawdata`
	Il bucket S3 è configurato per eseguire la funzione di indice Lambda ogni volta che i dati vengono caricati in modo che il documento venga indicizzato in Elasticsearch.
		Architetto del cloud, amministratore del cloud

Cerca dati dalla console Kibana.	Sulla console Kibana, esegui la seguente query: `GET tenant-data/_search` Questa query mostra tutti i documenti indicizzati in Elasticsearch. In questo caso, dovresti vedere due documenti separati per Tenant-1 e Tenant-2.	Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste

Prova l'API di ricerca da API Gateway.	1. Nella console API Gateway, apri l'API di ricerca, scegli il `GET` metodo all'interno della risorsa di ricerca, quindi scegli Test.
	2. Nella finestra di test, fornisci la seguente stringa di query (con distinzione tra maiuscole e minuscole) per l'ID del tenant, quindi scegli Test.
	`TenantId=Tenant-1`
	La funzione Lambda invia una query a OpenSearch Service che filtra il documento del tenant in base alla sicurezza a livello di documento. Il metodo restituisce il documento che appartiene a Tenant-1.
	3. Cambia la stringa di interrogazione in:
	`TenantId=Tenant-2`
	Questa query restituisce il documento che appartiene a Tenant-2.
		Architetto del cloud, sviluppatore di app

Operazione	Description	Competenze richieste
	Per le illustrazioni delle schermate, vedere la sezione Informazioni aggiuntive.	
Eliminare le risorse.	Pulisci tutte le risorse che hai creato per evitare addebiti aggiuntivi sul tuo account.	AWS DevOps, architetto cloud, amministratore cloud

Risorse correlate

- AWS SDK per Python (Boto)
- AWS Lambda documentazione
- Documentazione API Gateway
- Documentazione Amazon S3
- Documentazione OpenSearch del servizio Amazon
  - Controllo granulare degli accessi in Amazon Service OpenSearch
  - Creazione di un'applicazione di ricerca con Amazon OpenSearch Service
  - Avvio dei domini Amazon OpenSearch Service all'interno di un VPC

Informazioni aggiuntive

Modelli di partizionamento dei dati

Esistono tre modelli di partizionamento dei dati comuni utilizzati nei sistemi multi-tenant: silo, pool e hybrid. Il modello scelto dipende dalla conformità, dalla rumorosità dei sistemi vicini, dalle operazioni e dalle esigenze di isolamento dell'ambiente.

Modello Silo

Nel modello a silo, i dati di ciascun inquilino vengono archiviati in un'area di archiviazione distinta in cui non vi è alcuna combinazione dei dati del tenant. È possibile utilizzare due approcci per implementare il modello a silo con OpenSearch Service: dominio per tenant e indice per tenant.

- Dominio per tenant: puoi utilizzare un dominio di OpenSearch servizio separato (sinonimo di cluster Elasticsearch) per tenant. L'inserimento di ogni tenant nel proprio dominio offre tutti i vantaggi

associati alla presenza di dati in un costrutto autonomo. Tuttavia, questo approccio introduce sfide di gestione e agilità. La sua natura distribuita rende più difficile l'aggregazione e la valutazione dello stato operativo e dell'attività degli inquilini. Si tratta di un'opzione costosa che richiede che ogni dominio di OpenSearch servizio disponga almeno di tre nodi master e due nodi di dati per i carichi di lavoro di produzione.

- Indice per tenant: è possibile inserire i dati dei tenant in indici separati all'interno di un cluster di servizi. OpenSearch Con questo approccio, si utilizza un identificatore del tenant quando si crea e si assegna un nome all'indice, anteponendo l'identificatore del tenant al nome dell'indice. L'approccio dell'indice per tenant consente di raggiungere gli obiettivi dei silo senza introdurre un cluster completamente separato per ogni tenant. Tuttavia, se il numero di indici aumenta, si potrebbe verificare una pressione sulla memoria, poiché questo approccio richiede più shard e il nodo master deve gestire una maggiore allocazione e ribilanciamento.

Isolamento nel modello a silo: nel modello a silo, si utilizzano le policy IAM per isolare i domini o gli indici che contengono i dati di ciascun tenant. Queste politiche impediscono a un tenant di accedere ai dati di un altro tenant. Per implementare il modello di isolamento dei silo, è possibile creare una politica basata sulle risorse che controlli l'accesso alla risorsa del tenant. Si tratta spesso di una politica di accesso al dominio che specifica quali azioni un principale può eseguire sulle risorse secondarie del dominio, inclusi gli indici Elasticsearch e. APIs Con le policy basate sull'identità IAM, puoi specificare azioni consentite o negate sul dominio, sugli indici o all'interno del Servizio. APIs OpenSearch L'`Action`elemento di una policy IAM descrive l'azione o le azioni specifiche consentite o negate dalla policy e specifica gli account, gli `Principal` utenti o i ruoli interessati.

La seguente policy di esempio concede al Tenant-1 l'accesso completo (come specificato da`es:*`) solo alle risorse secondarie del dominio. `tenant-1` La fine `/*` dell'`Resource`elemento indica che questa politica si applica alle risorse secondarie del dominio, non al dominio stesso. Quando questa politica è in vigore, i tenant non sono autorizzati a creare un nuovo dominio o modificare le impostazioni su un dominio esistente.

```
{
   "Version": "2012-10-17",
```

```
   "Statement": [
      {
         "Effect": "Allow",
         "Principal": {
            "AWS": "arn:aws:iam::<aws-account-id>:user/Tenant-1"
         },
         "Action": "es:*",
         "Resource": "arn:aws:es:<Region>:<account-id>:domain/tenant-1/*"
      }
   ]
}
```

Per implementare il modello di silo tenant per index, è necessario modificare questa politica di esempio per limitare ulteriormente Tenant-1 all'indice o agli indici specificati, specificando il nome dell'indice. La seguente politica di esempio limita Tenant-1 all'indice. `tenant-index-1`

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Principal": {
            "AWS": "arn:aws:iam::123456789012:user/Tenant-1"
         },
         "Action": "es:*",
         "Resource": "arn:aws:es:<Region>:<account-id>:domain/test-domain/tenant-
index-1/*"
      }
   ]
}
```

Modello di piscina

Nel modello di pool, tutti i dati dei tenant vengono archiviati in un indice all'interno dello stesso dominio. L'identificatore del tenant è incluso nei dati (documento) e utilizzato come chiave di partizione, in modo da poter determinare quali dati appartengono a quale tenant. Questo modello riduce il sovraccarico di gestione. Il funzionamento e la gestione dell'indice raggruppato sono più semplici ed efficienti rispetto alla gestione di più indici. Tuttavia, poiché i dati dei tenant vengono combinati all'interno dello stesso indice, si perde il naturale isolamento dei tenant fornito dal modello a silo. Questo approccio potrebbe inoltre ridurre le prestazioni a causa dell'effetto Noisy Neighbor.

Isolamento dei tenant nel modello pool: in generale, l'isolamento dei tenant è difficile da implementare nel modello pool. Il meccanismo IAM utilizzato con il modello a silo non consente di descrivere l'isolamento in base all'ID del tenant memorizzato nel documento.

Un approccio alternativo consiste nell'utilizzare il supporto per il controllo degli accessi a grana fine (FGAC) fornito da Open Distro for Elasticsearch. FGAC consente di controllare le autorizzazioni a livello di indice, documento o campo. Con ogni richiesta, FGAC valuta le credenziali dell'utente e autentica l'utente o nega l'accesso. Se FGAC autentica l'utente, recupera tutti i ruoli mappati a quell'utente e utilizza il set completo di autorizzazioni per determinare come gestire la richiesta.

Per ottenere l'isolamento richiesto nel modello in pool, è possibile utilizzare la sicurezza a livello di documento, che consente di limitare un ruolo a un sottoinsieme di documenti in un indice. Il seguente ruolo di esempio limita le query a Tenant-1. Applicando questo ruolo a Tenant-1, è possibile ottenere l'isolamento necessario.

```
{
   "bool": {
     "must": {
       "match": {
         "tenantId": "Tenant-1"
       }
     }
   }
 }
```

Modello ibrido

Il modello ibrido utilizza una combinazione dei modelli a silo e pool nello stesso ambiente per offrire esperienze uniche a ciascun livello di tenant (ad esempio i livelli gratuito, standard e premium). Ogni livello segue lo stesso profilo di sicurezza utilizzato nel modello pool.

Isolamento dei tenant nel modello ibrido: nel modello ibrido, si segue lo stesso profilo di sicurezza del modello pool, dove l'utilizzo del modello di sicurezza FGAC a livello di documento forniva l'isolamento dei tenant. Sebbene questa strategia semplifichi la gestione dei cluster e offra agilità, complica altri aspetti dell'architettura. Ad esempio, il codice richiede una complessità aggiuntiva per determinare quale modello è associato a ciascun tenant. È inoltre necessario assicurarsi che le query relative a un solo tenant non saturino l'intero dominio e compromettano l'esperienza degli altri tenant.

Test in API Gateway

Finestra di test per la query Tenant-1

Finestra di test per la query Tenant-2

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Distribuisci applicazioni multi-stack utilizzando AWS CDK con TypeScript

Dott. Rahul Sharad Gaikwad, Amazon Web Services

Riepilogo

Questo modello fornisce un step-by-step approccio per la distribuzione di applicazioni su Amazon Web Services (AWS) utilizzando AWS Cloud Development Kit (AWS CDK) con TypeScript. Ad esempio, il pattern implementa un'applicazione di analisi in tempo reale senza server.

Il pattern crea e distribuisce applicazioni stack annidate. Lo stack AWS principale chiama gli CloudFormation stack secondari, o annidati. Ogni stack secondario crea e distribuisce le risorse AWS definite nello stack. CloudFormation AWS CDK Toolkit, il comando dell'interfaccia a riga di comando (CLI)`cdk`, è l'interfaccia principale per gli stack. CloudFormation

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Cloud privato virtuale (VPC) e sottoreti esistenti
- AWS CDK Toolkit installato e configurato
- Un utente con autorizzazioni di amministratore e un set di chiavi di accesso.
- Node.js
- Interfaccia a riga di comando di AWS (CLI AWS)

Limitazioni

- Poiché AWS CDK utilizza AWS CloudFormation, le applicazioni AWS CDK sono soggette a quote di CloudFormation servizio. Per ulteriori informazioni, consulta AWS CloudFormation quotas.

Versioni del prodotto

Questo modello è stato creato e testato utilizzando i seguenti strumenti e versioni.

- Kit di strumenti CDK AWS 1.83.0
- Node.js 14.13.0

- npm 7.0.14

Il modello dovrebbe funzionare con qualsiasi versione di AWS CDK o npm. Tieni presente che le versioni di Node.js dalla 13.0.0 alla 13.6.0 non sono compatibili con AWS CDK.

Architecture

Stack tecnologico Target

- Console AWS Amplify
- Gateway Amazon API
- AWS CDK
- Amazon CloudFront
- Amazon Cognito
- Amazon DynamoDB
- Amazon Data Firehose
- Flusso di dati Amazon Kinesis
- AWS Lambda
- Amazon Simple Storage Service (Amazon S3)

Architettura Target

Il diagramma seguente mostra la distribuzione di applicazioni a stack multiplo utilizzando AWS CDK con. TypeScript

Il diagramma seguente mostra l'architettura dell'applicazione serverless in tempo reale di esempio.

Tools (Strumenti)

Strumenti

- La console AWS Amplify è il centro di controllo per le distribuzioni complete di applicazioni web e mobili in AWS. L'hosting di Amplify Console offre un flusso di lavoro basato su git per l'hosting di

app web serverless fullstack con distribuzione continua. L'interfaccia utente di amministrazione è un'interfaccia visiva per gli sviluppatori web e mobili di frontend per creare e gestire i backend di app al di fuori della console AWS.

- Amazon API Gateway è un servizio AWS per la creazione, la pubblicazione, la manutenzione, il monitoraggio e la protezione di REST, HTTP e WebSocket APIs su qualsiasi scala.
- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software che aiuta a definire e fornire l'infrastruttura cloud AWS in codice.
- AWS CDK Toolkit è un kit di sviluppo cloud a riga di comando che ti aiuta a interagire con la tua app AWS CDK. Il comando `cdk` CLI è lo strumento principale per interagire con l'app AWS CDK. Esegue la tua app, interroga il modello di applicazione che hai definito e produce e distribuisce i CloudFormation modelli AWS generati dal CDK AWS.
- Amazon CloudFront è un servizio Web che accelera la distribuzione di contenuti Web statici e dinamici, come .html, .css, .js e file di immagine. CloudFront distribuisce i tuoi contenuti attraverso una rete mondiale di data center denominati edge location per ridurre la latenza e migliorare le prestazioni.
- Amazon Cognito fornisce autenticazione, autorizzazione e gestione degli utenti per le tue app Web e mobili. I tuoi utenti possono accedere direttamente o tramite terze parti.
- Amazon DynamoDB è un servizio di database NoSQL completamente gestito che offre prestazioni veloci e prevedibili con una scalabilità perfetta.
- Amazon Data Firehose è un servizio completamente gestito per la distribuzione di dati di streaming in tempo reale a destinazioni come Amazon S3, Amazon Redshift, OpenSearch Amazon Service, Splunk e qualsiasi endpoint HTTP o endpoint HTTP personalizzato di proprietà di provider di servizi terzi supportati.
- Amazon Kinesis Data Streams è un servizio per la raccolta e l'elaborazione di grandi flussi di record di dati in tempo reale.
- AWS Lambda è un servizio di elaborazione che supporta l'esecuzione di codice senza effettuare il provisioning o la gestione di server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo. Verrà addebitato soltanto il tempo di calcolo consumato e non verrà addebitato alcun costo quando il codice non è in esecuzione.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Codice

Il codice per questo modello è allegato.

Epiche

Installa AWS CDK Toolkit

Operazione	Description	Competenze richieste
Installa AWS CDK Toolkit.	Per installare AWS CDK Toolkit a livello globale, esegui il seguente comando. `npm install -g aws-cdk`	DevOps
Verifica la versione.	Per verificare la versione di AWS CDK Toolkit, esegui il comando seguente. `cdk --version`	DevOps

Configura le credenziali AWS

Operazione	Description	Competenze richieste

Configura le credenziali.	Per configurare le credenziali, esegui il `aws configure` comando e segui le istruzioni.
	`$aws configure`
	`AWS Access Key ID`
	`[None]:`
	`AWS Secret Access Key`
	`[None]: your_secr`
	`et_access_key`
	`Default region name`
	`[None]:`
	`Default output format`
	`[None]:`
		DevOps

Scarica il codice del progetto

Operazione	Description	Competenze richieste
Scarica il codice del progetto allegato.	Per ulteriori informazioni sulla directory e sulla struttura dei file, consulta la sezione Informazioni aggiuntive.	DevOps

Esegui il bootstrap dell'ambiente AWS CDK

Operazione	Description	Competenze richieste
Avvia l'ambiente.	Per distribuire il CloudFormation modello AWS nell'account e nella regione AWS che desideri utilizzare, esegui il comando seguente. `cdk bootstrap <account>/<Region>` Per ulteriori informazioni, consulta la documentazione di AWS.	DevOps

Crea e distribuisci il progetto

Operazione	Description	Competenze richieste
Compilare il progetto.	Per creare il codice del progetto, esegui il `npm run build` comando.	DevOps

Operazione	Description	Competenze richieste
Distribuisci il progetto	Per distribuire il codice del progetto, esegui il `cdk deploy` comando.	

Verifica gli output

Operazione	Description	Competenze richieste
Verifica la creazione dello stack.	Nella Console di gestione AWS, scegli CloudFormation. Negli stack del progetto, verifica che siano stati creati uno stack principale e due stack secondari.	DevOps

Eseguire il test dell'applicazione

Operazione	Description	Competenze richieste
Invia dati a Kinesis Data Streams.	Configura il tuo account AWS per inviare dati a Kinesis Data Streams utilizzando Amazon Kinesis Data Generator (KDG). Per ulteriori informazioni, consulta Amazon Kinesis Data Generator.	DevOps
Crea un utente Amazon Cognito.	Per creare un utente Amazon Cognito, scarica il modello cognito-setup.json CloudFormation dalla sezione Crea un utente Amazon Cognito della pagina di aiuto di Kinesis Data Generator. Avvia il modello,	DevOps

Operazione	Description	Competenze richieste
	quindi inserisci il nome utente e la password di Amazon Cognito. La scheda Output elenca l'URL di Kinesis Data Generator.	
Accedi a Kinesis Data Generator	Per accedere a KDG, utilizza le credenziali di Amazon Cognito che hai fornito e l'URL di Kinesis Data Generator.	DevOps
Testare l'applicazione.	In KDG, in Record template, Template 1, incolla il codice di test dalla sezione Informazioni aggiuntive e scegli Invia dati.	DevOps
Gateway API di prova.	Dopo che i dati sono stati inseriti, prova API Gateway utilizzando il `GET` metodo per recuperare i dati.	DevOps

Risorse correlate

Riferimenti

- Kit di sviluppo AWS per il cloud
- CDK AWS su GitHub
- Lavorare con stack annidati
- Esempio di AWS: analisi in tempo reale senza server

Informazioni aggiuntive

Dettagli di directory e file

Questo modello imposta le seguenti tre pile.

- `parent-cdk-stack.ts`— Questo stack funge da stack principale e chiama le due applicazioni secondarie come stack annidati.
- `real-time-analytics-poc-stack.ts`— Questo stack annidato contiene l'infrastruttura e il codice dell'applicazione.
- `real-time-analytics-web-stack.ts`— Questo stack annidato contiene solo il codice statico dell'applicazione Web.

File importanti e relative funzionalità

- `bin/real-time-analytics-poc.ts`— Punto di ingresso dell'applicazione AWS CDK. Carica tutti gli stack definiti in. `lib/`
- `lib/real-time-analytics-poc-stack.ts`— Definizione dello stack dell'applicazione AWS CDK ()`real-time-analytics-poc`.
- `lib/real-time-analytics-web-stack.ts`— Definizione dello stack dell'applicazione AWS CDK ()`real-time-analytics-web-stack`.
- `lib/parent-cdk-stack.ts`— Definizione dello stack dell'applicazione AWS CDK ()`parent-cdk`.
- `package.json`— manifesto del modulo npm, che include il nome, la versione e le dipendenze dell'applicazione.
- `package-lock.json`— Gestito da npm.
- `cdk.json`— Toolkit per l'esecuzione dell'applicazione.
- `tsconfig.json`— La TypeScript configurazione del progetto.
- `.gitignore`— Elenco di file che Git dovrebbe escludere dal controllo del codice sorgente.
- `node_modules`— Gestito da npm; include le dipendenze del progetto.

La seguente sezione di codice nello stack principale chiama le applicazioni secondarie come stack CDK AWS annidati.

```
import * as cdk from '@aws-cdk/core';
import { Construct, Stack, StackProps } from '@aws-cdk/core';
import { RealTimeAnalyticsPocStack } from './real-time-analytics-poc-stack';
import { RealTimeAnalyticsWebStack } from './real-time-analytics-web-stack';
```

```
export class CdkParentStack extends Stack {
  constructor(scope: Construct, id: string, props?: StackProps) {
    super(scope, id, props);


    new RealTimeAnalyticsPocStack(this, 'RealTimeAnalyticsPocStack');
    new RealTimeAnalyticsWebStack(this, 'RealTimeAnalyticsWebStack');
  }
}
```

Codice per i test

```
session={{date.now('YYYYMMDD')}}|sequence={{date.now('x')}}|
reception={{date.now('x')}}|instrument={{random.number(9)}}|
l={{random.number(20)}}|price_0={{random.number({"min":10000,
 "max":30000})}}|price_1={{random.number({"min":10000, "max":30000})}}|
price_2={{random.number({"min":10000, "max":30000})}}|
price_3={{random.number({"min":10000, "max":30000})}}|
price_4={{random.number({"min":10000, "max":30000})}}|
price_5={{random.number({"min":10000, "max":30000})}}|
price_6={{random.number({"min":10000, "max":30000})}}|
price_7={{random.number({"min":10000, "max":30000})}}|
price_8={{random.number({"min":10000, "max":30000})}}|
```

Test dell'API Gateway

Sulla console API Gateway, prova API Gateway utilizzando il `GET` metodo.

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Automatizza la distribuzione di applicazioni annidate utilizzando AWS SAM

Dott. Rahul Sharad Gaikwad, Ishwar Chauthaiwale, Dmitry Gulin e Tabby Ward, Amazon Web Services

Riepilogo

Su Amazon Web Services (AWS), AWS Serverless Application Model (AWS SAM) Serverless Application Model (AWS) Serverless Application Model (AWS) Serverless Application Model (AWS SAM) è un framework open source che fornisce una sintassi abbreviata per esprimere funzioni APIs, database e mappature delle sorgenti degli eventi. Con solo poche righe per ogni risorsa, puoi definire l'applicazione che desideri e modellarla utilizzando YAML. Durante la distribuzione, SAM trasforma ed espande la sintassi SAM in sintassi AWS CloudFormation che puoi usare per creare applicazioni serverless più velocemente.

AWS SAM semplifica lo sviluppo, la distribuzione e la gestione di applicazioni serverless sulla piattaforma AWS. Fornisce un framework standardizzato, una distribuzione più rapida, funzionalità di test locali, gestione delle risorse, perfetta integrazione con gli strumenti di sviluppo e una community di supporto. Queste caratteristiche lo rendono uno strumento prezioso per creare applicazioni serverless in modo efficiente ed efficace.

Questo modello utilizza modelli AWS SAM per automatizzare la distribuzione di applicazioni annidate. Un'applicazione annidata è un'applicazione all'interno di un'altra applicazione. Le applicazioni principali chiamano le proprie applicazioni secondarie. Si tratta di componenti liberamente accoppiati di un'architettura serverless.

Utilizzando applicazioni annidate, puoi creare rapidamente architetture serverless altamente sofisticate riutilizzando servizi o componenti creati e gestiti in modo indipendente ma composti utilizzando AWS SAM e Serverless Application Repository. Le applicazioni annidate ti aiutano a creare applicazioni più potenti, a evitare il lavoro duplicato e a garantire la coerenza e le migliori pratiche tra i tuoi team e le tue organizzazioni. Per dimostrare le applicazioni annidate, il pattern distribuisce un esempio di applicazione AWS serverless per il carrello degli acquisti.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un cloud privato virtuale (VPC) e sottoreti esistenti

- Un ambiente di sviluppo integrato come Visual Studio Code (per ulteriori informazioni, consulta Tools to Build on AWS)
- Libreria Python wheel installata utilizzando pip install wheel, se non è già installata

Limitazioni

- Il numero massimo di applicazioni che possono essere annidate in un'applicazione serverless è 200.
- Il numero massimo di parametri per un'applicazione annidata può essere 60.

Versioni del prodotto

- Questa soluzione è basata sull'interfaccia a riga di comando AWS SAM (AWS SAM CLI) versione 1.21.1, ma questa architettura dovrebbe funzionare con le versioni successive dell'interfaccia a riga di comando AWS SAM.

Architecture

Stack tecnologico Target

- Gateway Amazon API
- AWS SAM
- Amazon Cognito
- Amazon DynamoDB
- AWS Lambda
- Coda Amazon Simple Queue Service (Amazon SQS)

Architettura di destinazione

Il diagramma seguente mostra come vengono effettuate le richieste degli utenti ai servizi di acquisto tramite chiamata APIs. La richiesta dell'utente, incluse tutte le informazioni necessarie, viene inviata ad Amazon API Gateway e all'autorizzatore Amazon Cognito, che esegue i meccanismi di autenticazione e autorizzazione per. APIs

Quando un elemento viene aggiunto, eliminato o aggiornato in DynamoDB, un evento viene inserito in DynamoDB Streams, che a sua volta avvia una funzione Lambda. Per evitare l'eliminazione

immediata dei vecchi elementi come parte di un flusso di lavoro sincrono, i messaggi vengono inseriti in una coda SQS, che avvia una funzione di lavoro per eliminare i messaggi.

In questa configurazione della soluzione, AWS SAM CLI funge da interfaccia per gli stack CloudFormation AWS. I modelli AWS SAM distribuiscono automaticamente applicazioni annidate. Il modello SAM principale chiama i modelli secondari e lo stack principale distribuisce gli CloudFormation stack secondari. Ogni stack secondario crea le risorse AWS definite nei modelli AWS SAM CloudFormation .

1. Crea e distribuisci gli stack.
2. Lo CloudFormation stack di autenticazione contiene Amazon Cognito.
3. Lo CloudFormation stack di prodotti contiene una funzione Lambda e Amazon API Gateway
4. Lo CloudFormation stack Shopping contiene una funzione Lambda, Amazon API Gateway, la coda SQS e il database Amazon DynamoDB.

Tools (Strumenti)

Strumenti

- Amazon API Gateway ti aiuta a creare, pubblicare, gestire, monitorare e proteggere REST, HTTP e WebSocket APIs su qualsiasi scala.
- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account e regioni AWS.
- Amazon Cognito fornisce autenticazione, autorizzazione e gestione degli utenti per app Web e mobili.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- AWS Serverless Application Model (AWS SAM) Serverless Application Model (AWS SAM) è un framework open source che ti aiuta a creare applicazioni serverless nel cloud AWS.
- Amazon Simple Queue Service (Amazon SQS) fornisce una coda ospitata sicura, durevole e disponibile che ti aiuta a integrare e disaccoppiare sistemi e componenti software distribuiti.

Codice

Il codice per questo modello è disponibile nel repository GitHub AWS SAM Nested Stack Sample.

Epiche

Installa AWS SAM CLI

Operazione	Description	Competenze richieste
Installa AWS SAM CLI.	Per installare AWS SAM CLI, consulta le istruzioni nella documentazione di AWS SAM.	DevOps ingegnere

Configura le credenziali AWS.	Per impostare le credenziali AWS in modo che la CLI di AWS SAM possa effettuare chiamate ai servizi AWS per tuo conto, esegui `aws configure` il comando e segui le istruzioni.
	`$aws configure`
	`AWS Access Key ID [None]: <your_access_key_id>`
	`AWS Secret Access Key [None]: your_secret_access_key`
	`Default region name [None]:`
	`Default output format [None]:`
	Per ulteriori informazioni sulla configurazione delle credenziali, consulta Autenticazione e credenziali di accesso.
		DevOps ingegnere

Inizializza il progetto AWS SAM

Operazione	Description	Competenze richieste

Clona il repository di codice AWS SAM.	1. Clona il repository di esempio aws sam nested stack per questo pattern inserendo il seguente comando.
	`git clone https://github.com/aws-samples/aws-sam-nested-stack-sample.git`
	2. Naviga nella directory clonata inserendo il seguente comando.
	`cd aws-sam-nested-stack-sample`
		DevOps ingegnere

Implementa modelli per inizializzare il progetto.	Per inizializzare il progetto, esegui il comando. `SAM init` Quando viene richiesto di scegliere una fonte per il modello, scegliete. `Custom Template Location`	DevOps ingegnere

Compila e crea il codice del modello SAM

Operazione	Description	Competenze richieste
Esamina i modelli di applicazione AWS SAM.	Esamina i modelli per le applicazioni annidate. Questo esempio utilizza i seguenti	DevOps ingegnere

Operazione	Description	Competenze richieste
	modelli di applicazioni annidate: • `auth.yaml` — Questo modello configura risorse relative all'autenticazione , come Amazon Cognito e AWS Systems Manager Parameter Store. • `product-mock.yaml` — Questo modello distribuisce risorse relative al prodotto, come le funzioni Lambda e Amazon API Gateway. • `shoppingcart-service.yaml` — Questo modello configura risorse relative al carrello della spesa, come AWS Identity and Access Management (IAM), tabelle DynamoDB e funzioni Lambda.	
Rivedi il modello principale.	Esamina il modello che richiamerà i modelli di applicazione annidati. In questo esempio, il modello principale è. `template.yml` Tutte le applicazioni separate sono annidate nell'unico modello `template.yml` principale.	DevOps ingegnere

Operazione	Description	Competenze richieste
Compila e crea il codice modello AWS SAM.	Utilizzando l'AWS SAM CLI, esegui il comando seguente. `sam build`	DevOps ingegnere

Implementa il modello AWS SAM

Operazione	Description	Competenze richieste
Implementa le applicazioni.	Per avviare il codice modello SAM che crea gli CloudFormation stack di applicazioni annidate e distribuisce il codice nell'ambiente AWS, esegui il comando seguente. `sam deploy --guided --stack-name shopping-cart-nested-stack --capabilities CAPABILITY_IAM CAPABILITY_AUTO_EXPAND` Il comando richiederà alcune domande. Rispondi a tutte le domande cony.	DevOps ingegnere

Verifica della distribuzione

Operazione	Description	Competenze richieste
Verifica le pile.	Per esaminare gli CloudFormation stack AWS e le risorse	DevOps ingegnere

Operazione	Description	Competenze richieste
	AWS definiti nei modelli AWS SAM, procedi come segue: 1. Accedi alla Console di gestione AWS e accedi alla CloudFormationconsole. 2. Verifica che gli stack principale e secondario siano elencati. In questo esempio, `sam-shopping-cart` è lo stack principale che chiama gli stack annidati Auth, Product e Shopping. Lo stack di prodotti fornisce il link URL del Product API Gateway come output.	

Risorse correlate

Riferimenti

- Modello di applicazioni serverless AWS (AWS SAM)
- AWS SAM su GitHub
- Microservizio Serverless Shopping Cart (applicazione di esempio AWS)

Tutorial e video

- Crea un'app serverless
- AWS Online Tech Talks: creazione e implementazione di applicazioni serverless con AWS SAM

Informazioni aggiuntive

Dopo che tutto il codice è a posto, l'esempio ha la seguente struttura di directory:

- sam_stacks — Questa cartella contiene il layer. `shared.py` Un layer è un archivio di file che contiene librerie, un runtime personalizzato o altre dipendenze. Con i livelli, puoi utilizzare le librerie nella tua funzione senza doverle includere in un pacchetto di distribuzione.
- product-mock-service— Questa cartella contiene tutte le funzioni e i file Lambda relativi al prodotto.
- shopping-cart-service— Questa cartella contiene tutte le funzioni e i file Lambda relativi agli acquisti.

Implementa l'isolamento dei tenant SaaS per Amazon S3 utilizzando un distributore automatico di token AWS Lambda

Tabby Ward, Thomas Davis e Sravan Periyathambi, Amazon Web Services

Riepilogo

Le applicazioni SaaS multitenant devono implementare sistemi per garantire il mantenimento dell'isolamento dei tenant. Quando memorizzi i dati dei tenant sulla stessa AWS risorsa, ad esempio quando più tenant archiviano i dati nello stesso bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3), devi assicurarti che l'accesso tra tenant non possa avvenire. I distributori automatici di token () sono un modo per garantire l'isolamento dei dati dei tenant. TVMs Queste macchine forniscono un meccanismo per ottenere token e al contempo astrarre la complessità del modo in cui questi token vengono generati. Gli sviluppatori possono utilizzare una TVM senza avere una conoscenza dettagliata di come produce i token.

Questo modello implementa una TVM utilizzando. AWS Lambda Il TVM genera un token costituito da credenziali temporanee del servizio token di sicurezza (STS) che limitano l'accesso ai dati di un singolo tenant SaaS in un bucket S3.

TVMs, e il codice fornito con questo modello, vengono in genere utilizzati con attestazioni derivate da JSON Web Tokens (JWTs) per associare le richieste di AWS risorse a una policy tenant-scoped (IAM). AWS Identity and Access Management È possibile utilizzare il codice in questo modello come base per implementare un'applicazione SaaS che genera credenziali STS temporanee e con ambito basate sulle affermazioni fornite in un token JWT.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- AWS Command Line Interface (AWS CLI) versione 1.19.0 o successiva, installata e configurata su macOS, Linux o Windows. In alternativa, è possibile utilizzare la AWS CLI versione 2.1 o successiva.

Limitazioni

- Questo codice viene eseguito in Java e attualmente non supporta altri linguaggi di programmazione.

- L'applicazione di esempio non include il supporto AWS interregionale o di disaster recovery (DR).
- Questo modello dimostra come una Lambda TVM per un'applicazione SaaS possa fornire un accesso mirato ai tenant. Questo modello non è destinato all'uso in ambienti di produzione senza ulteriori test di sicurezza come parte dell'applicazione o del caso d'uso specifici.

Architecture

Stack tecnologico Target

- AWS Lambda
- Simple Storage Service (Amazon S3)
- IAM
- AWS Security Token Service (AWS STS)

Architettura di destinazione

Tools (Strumenti)

Servizi AWS

- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS Security Token Service (AWS STS) consente di richiedere credenziali temporanee con privilegi limitati per gli utenti.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Codice

Il codice sorgente di questo pattern è disponibile come allegato e include i seguenti file:

- `s3UploadSample.jar`fornisce il codice sorgente per una funzione Lambda che carica un documento JSON in un bucket S3.
- `tvm-layer.zip`fornisce una libreria Java riutilizzabile che fornisce un token (credenziali temporanee STS) per la funzione Lambda per accedere al bucket S3 e caricare il documento JSON.
- `token-vending-machine-sample-app.zip`fornisce il codice sorgente usato per creare questi artefatti e le istruzioni di compilazione.

Per utilizzare questi file, seguite le istruzioni riportate nella sezione successiva.

Epiche

Determina i valori delle variabili

Operazione	Description	Competenze richieste
Determina i valori delle variabili.	L'implementazione di questo modello include diversi nomi di variabili che devono essere usati in modo coerente. Determina i valori da utilizzare per ogni variabile e fornisci quel valore quando richiesto nei passaggi successivi. • `<AWS Account ID>`— L'ID dell'account a 12 cifre associato al modello in cui Account AWS stai implementando questo modello. Per informazioni su come trovare il tuo Account AWS ID, consulta Usare un alias per il tuo Account AWS ID nella documentazione IAM.	Amministratore cloud

Operazione	Description	Competenze richieste
	• `<AWS Region>`— Il Regione AWS modo in cui stai implementando questo modello. Per ulteriori informazioni Regioni AWS, consulta Regioni e zone di disponibilità sul AWS sito Web. • `<sample-tenant-name>` — Il nome di un inquilino da utilizzare nell'applicazione. Si consiglia di utilizzare solo caratteri alfanumerici in questo valore per semplicità, ma è possibile utilizzare qualsiasi nome valido per una chiave oggetto S3. • `<sample-tvm-role-name>` — Il nome del ruolo IAM associato alla funzione Lambda che esegue TVM e l'applicazione di esempio. Il nome del ruolo è una stringa composta da caratteri alfanumerici maiuscoli e minuscoli senza spazi. È inoltre possibile includere uno qualsiasi dei seguenti caratteri: trattino basso (_), segno più (+), segno uguale (=), virgola (,), punto (.), segno chiocciola (@) e trattino (-). Il nome del ruolo	

Operazione	Description	Competenze richieste
	deve essere univoco all'interno dell'account. • `<sample-app-role-name>` – Il nome del ruolo IAM assunto dalla funzione Lambda quando genera credenziali STS temporanee e con ambito. Il nome del ruolo è una stringa composta da caratteri alfanumerici maiuscoli e minuscoli senza spazi. È inoltre possibile includere uno qualsiasi dei seguenti caratteri: trattino basso (_), segno più (+), segno uguale (=), virgola (,), punto (.), segno chiocciola (@) e trattino (-). Il nome del ruolo deve essere univoco all'interno dell'account. • `<sample-app-function-name>` – Il nome della funzione Lambda. Si tratta di una stringa che può contenere fino a 64 caratteri. • `<sample-app-bucket-name>` – Il nome di un bucket S3 a cui è necessario accedere con autorizzazioni limitate a un tenant specifico. Nomi dei bucket S3:	

Operazione	Description	Competenze richieste
	• Devono contenere da 3 a 63 caratteri. • Deve essere composto solo da lettere minuscole , numeri, punti (.) e trattini (-). • Deve iniziare e terminare con una lettera o un numero. • Non devono avere il formato di un indirizzo IP (ad esempio, 192.168.5 .4). • Deve essere univoco all'interno di una partizion e. Una partizione è un raggruppamento di regioni. AWS attualmen te ha tre partizioni: `aws` per le regioni standard, `aws-cn` per le regioni della Cina e `aws-us-go v` per AWS GovCloud (US) Regions.	

Creare un bucket S3

Operazione	Description	Competenze richieste
Crea un bucket S3 per l'applicazione di esempio.	Usa il seguente AWS CLI comando per creare un bucket S3. Fornisci il `<sample-a`	Amministratore cloud

Operazione	Description	Competenze richieste
	pp-bucket-name> valore nel frammento di codice: `aws s3api create-bucket --bucket <sample-app-bucket-name>` L'applicazione di esempio Lambda carica i file JSON in questo bucket.	

Crea il ruolo e la policy di IAM TVM

Operazione	Description	Competenze richieste
Crea un ruolo TVM.	Utilizza uno dei seguenti AWS CLI comandi per creare un ruolo IAM. Fornisci il <sample-tvm-role-name> valore nel comando. Per le shell macOS o Linux: `aws iam create-role \ --role-name <sample-tvm-role-name> \ --assume-role-policy-document '{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [ "sts:AssumeRole"`	Amministratore cloud

Operazione	Description	Competenze richieste

```
            ],
            "Principa
l": {

 "Service": [

 "lambda.amazonaws.
com"
                ]
            },
            "Conditio
n": {

 "StringEquals": {

 "aws:SourceAccount":
 "<AWS Account ID>"
                }
            }
        }
    ]
}'
```

Per la riga di comando di Windows:

```
aws iam create-role ^
--role-name <sample-t
vm-role-name> ^
--assume-role-policy-
document "{\"Versi
on\": \"2012-10
-17\", \"Statement
\": [{\"Effect\":
 \"Allow\", \"Action\
": [\"sts:AssumeRole
\"], \"Principal
\": {\"Service\":
 [\"lambda.amazonaw
s.com\"]}, \"Conditi
on\": {\"StringEquals
```

Operazione	Description	Competenze richieste

	<pre>\": {\"aws:SourceAccou
	nt\": \"<AWS Account
	 ID>\"}}}]}"</pre>
	L'applicazione di esempio Lambda assume questo ruolo quando viene richiamata l'applicazione. La capacità di assumere il ruolo dell'applicazione con una policy mirata offre al codice autorizzazioni più ampie per accedere al bucket S3.	

Operazione	Description	Competenze richieste

Crea una politica di ruolo TVM in linea.	Utilizzate uno dei seguenti AWS CLI comandi per creare una policy IAM. Fornisci i `<sample-app-role-name>` valori `<sample-tvm-role-name>` **`<AWS Account ID>`**, e nel comando.
	
	Per le shell macOS o Linux:
	
	`aws iam put-role-policy \ --role-name <sample-tvm-role-name> \ --policy-name assume-app-role \ --policy-document '{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": "sts:AssumeRole", "Resource": "arn:aws:iam::<AWS Account ID>:role/<sample-app-role-name>" } ]}'`
	
	Per la riga di comando di Windows:
	
	`aws iam put-role-policy ^`
		Amministratore cloud

Operazione	Description	Competenze richieste
	`--role-name <sample-tvm-role-name> ^ --policy-name assume-app-role ^ --policy-documen t "{\"Version\": \"2012-10-17\", \"Statement\": [{\"Effect\": \"Allow\", \"Action\": \"sts:AssumeRole\", \"Resource\": \"arn:aws:iam::<AWS Account ID>:role/<sample-app-role-name>\"}]}"` Questa policy è associata al ruolo TVM. Fornisce al codice la capacità di assumere il ruolo dell'applicazione, che dispone di autorizzazioni più ampie per accedere al bucket S3.	

Operazione	Description	Competenze richieste

Allega la policy Lambda gestita.	Usa il AWS CLI comando seguente per allegare la policy `AWSLambdaBasicExecutionRole` IAM. Fornisci il `<sample-tvm-role-name>` valore nel comando:
	`aws iam attach-role-policy \`
	`--role-name <sample-tvm-role-name> \`
	`--policy-arn arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole`
	Per la riga di comando di Windows:
	`aws iam attach-role-policy ^`
	`--role-name <sample-tvm-role-name> ^`
	`--policy-arn arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole`
	Questa policy gestita è allegata al ruolo TVM per consentire a Lambda di inviare log ad Amazon. CloudWatch
		Amministratore cloud

Crea il ruolo e la policy dell'applicazione IAM

Operazione	Description	Competenze richieste
Crea il ruolo dell'applicazione.	Utilizza uno dei seguenti AWS CLI comandi per creare un ruolo IAM. Fornisci i `<sample-tvm-role-name>` valori `<sample-app-role-name>` `<AWS Account ID>`, e nel comando. Per le shell macOS o Linux: `aws iam create-role \ --role-name <sample-app-role-name> \ --assume-role-policy-document '{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "AWS": "arn:aws:iam::<AWS Account ID>:role/<sample-tvm-role-name>" }, "Action": "sts:AssumeRole" } ]}'` Per la riga di comando di Windows:	Amministratore del cloud

Operazione	Description	Competenze richieste
	`aws iam create-role ^ --role-name <sample-app-role-name> ^ --assume-role-policy-document "{\"Version\": \"2012-10-17\", \"Statement\": [{\"Effect\": \"Allow\",\"Principal\": {\"AWS\": \"arn:aws:iam::<AWS Account ID>:role/<sample-tvm-role-name>\"},\"Action\": \"sts:AssumeRole\" }]}"` L'applicazione di esempio Lambda assume questo ruolo con una policy mirata per ottenere l'accesso basato su tenant a un bucket S3.	

Operazione	Description	Competenze richieste
Crea una politica in linea per i ruoli delle applicazioni.	Utilizza uno dei seguenti AWS CLI comandi per creare una policy IAM. Fornisci i `<sample-app-bucket-name>` valori `<sample-app-role-name>` and nel comando. Per le shell macOS o Linux: `aws iam put-role-policy \ --role-name <sample-app-role-name> \ --policy-name s3-bucket-access \ --policy-document '{ "Version": "2012-10-17", "Statement": [{ "Effect": "Allow", "Action": ["s3:PutObject", "s3:GetObject", "s3:DeleteObject"], "Resource": "arn:aws:s3:::<sample-app-bucket-name>/*" }, { "Effect": "Allow",`	Amministratore cloud

Operazione	Description	Competenze richieste

	<pre>"Action":
	["s3:ListBucket"],
	"Resource
	": "arn:aws:s3:::<sam
	ple-app-bucket-name>"
	}
	]}'</pre>
	Per la riga di comando di Windows:
	<pre>aws iam put-role-policy
	^
	--role-name <sample-a
	pp-role-name> ^
	--policy-name s3-bucket
	-access ^
	--policy-documen
	t "{\"Version\":
	\"2012-10-17\",
	\"Statement\":
	[{\"Effect\": \"Allow
	\", \"Action\":
	[\"s3:PutObject\",
	\"s3:GetObject\",
	\"s3:DeleteObject\
	"], \"Resource\":
	\"arn:aws:s3:::<sa
	mple-app-bucket-na
	me>/*\"}, {\"Effect\":
	\"Allow\", \"Action\
	": [\"s3:ListBucket
	\"], \"Resource\":
	\"arn:aws:s3:::<sa
	mple-app-bucket-name>
	\"}]}"</pre>
	Questa politica è allegata al ruolo dell'applicazione. Fornisce un ampio accesso	

Operazione	Description	Competenze richieste
	agli oggetti nel bucket S3. Quando l'applicazione di esempio assume il ruolo, queste autorizzazioni sono limitate a un tenant specifico con la policy generata dinamicamente da TVM.	

Crea l'applicazione di esempio Lambda con TVM

Operazione	Description	Competenze richieste
Scarica i file sorgente compilati.	Scaricate i `tvm-layer.zip` file `s3UploadSample.jar` and, che sono inclusi come allegati. Il codice sorgente utilizzato per creare questi artefatti e le istruzioni di compilazione sono forniti in. `token-vending-machine-sample-app.zip`	Amministratore cloud
Crea il livello Lambda.	Utilizzate il seguente AWS CLI comando per creare un layer Lambda, che renda la TVM accessibile a Lambda. **Note** Se non state eseguendo questo comando dalla posizione in cui lo avete scaricato `tvm-layer.zip` , fornite	Amministratore cloud, sviluppatore di app

Operazione	Description	Competenze richieste
	il percorso corretto del parametro`tvm-layer.zip .--zip-file`	

```
aws lambda publish-layer-version \
--layer-name sample-token-vending-machine \
--compatible-runtimes java11 \
--zip-file fileb://tvm-layer.zip
```

Per la riga di comando di Windows:

```
aws lambda publish-layer-version ^
--layer-name sample-token-vending-machine ^
--compatible-runtimes java11 ^
--zip-file fileb://tvm-layer.zip
```

Questo comando crea un layer Lambda che contiene la libreria TVM riutilizzabile.

Operazione	Description	Competenze richieste
Creazione della funzione Lambda	Usa il AWS CLI comando seguente per creare una funzione Lambda. Fornisci i `<sample-app-role-name>` valori `<sample-app-function-name>` `<AWS Account ID><AWS Region>`, `<sample-tvm-role-name>` `<sample-app-bucket-name>` ,, e nel comando. **Note** Se non esegui questo comando dalla posizione in cui lo hai scaricato`s3UploadSample.jar` , fornisci il percorso corretto `s3UploadSample.jar` nel `--zip-file` parametro. `aws lambda create-function \ --function-name <sample-app-function-name> \ --timeout 30 \ --memory-size 256 \ --runtime java11 \ --role arn:aws:iam::<AWS Account`	Amministratore cloud, sviluppatore di app

Operazione	Description	Competenze richieste
	ID>:role/<sample-tvm-role-name> \ --handler com.amazon.aws.s3UploadSample.App \ --zip-file fileb://s3UploadSample.jar \ --layers arn:aws:lambda:<AWS Region>:<AWS Account ID>:layer:sample-token-vending-machine:1 \ --environment "Variables={S3_BUCKET=<sample-app-bucket-name>, ROLE=arn:aws:iam::<AWS Account ID>:role/<sample-app-role-name>}" Per la riga di comando di Windows: aws lambda create-function ^ --function-name <sample-app-function-name> ^ --timeout 30 ^ --memory-size 256 ^ --runtime java11 ^ --role arn:aws:iam::<AWS Account ID>:role/<sample-tvm-role-name> ^ --handler com.amazon.aws.s3UploadSample.App ^ --zip-file fileb://s3UploadSample.jar ^	

Operazione	Description	Competenze richieste
	`--layers arn:aws:lambda:<AWS Region>:<AWS Account ID>:layer:sample-token-vending-machine:1 ^ --environment "Variables={S3_BUCKET=<sample-app-bucket-name>,ROLE=arn:aws:iam::<AWS Account ID>:role/<sample-app-role-name>}"` Questo comando crea una funzione Lambda con il codice dell'applicazione di esempio e il livello TVM collegato. Imposta inoltre due variabili di ambiente: `S3_BUCKET` e. `ROLE` L'applicazione di esempio utilizza queste variabili per determinare il ruolo da assumere e il bucket S3 in cui caricare i documenti JSON.	

Prova l'applicazione di esempio e TVM

Operazione	Description	Competenze richieste
Richiama l'applicazione di esempio Lambda.	Utilizza uno dei seguenti AWS CLI comandi per avviare l'applicazione di esempio Lambda con il payload previsto. Fornisci i `<sample-tenant-name>` valori	Amministratore cloud, sviluppatore di app

Operazione	Description	Competenze richieste

	`<sample-app-function-name>` and nel comando.
	
	Per le shell macOS e Linux:
	
	`aws lambda invoke \ --function <sample-app-function-name> \ --invocation-type RequestResponse \ --payload '{"tenant": "<sample-tenant-name>"}' \ --cli-binary-format raw-in-base64-out response.json`
	
	Per la riga di comando di Windows:
	
	`aws lambda invoke ^ --function <sample-app-function-name> ^ --invocation-type RequestResponse ^ --payload "{\"tenant\": \"<sample-tenant-name>\"}" ^ --cli-binary-format raw-in-base64-out response.json`
	
	Questo comando richiama la funzione Lambda e restituisce il risultato in un `response.json` documento. In molti sistemi basati su Unix, è possibile passare `response.`	

Operazione	Description	Competenze richieste
	`json` a per `/dev/stdout` inviare i risultati direttamente alla shell senza creare un altro file. **Note** La modifica del `<sample-tenant-name>` valore nelle chiamate successiv e di questa funzione Lambda modifica la posizione del documento JSON e le autorizzazioni fornite dal token.	
Visualizza il bucket S3 per vedere gli oggetti creati.	Vai al bucket S3 (`<sample-app-bucket-name>`) che hai creato in precedenz a. Questo bucket contiene un prefisso di oggetto S3 con il valore di. `<sample-tenant-name>` Sotto quel prefisso, troverai un documento JSON denominat o con un UUID. Richiaman do più volte l'applicazione di esempio vengono aggiunti altri documenti JSON.	Amministratore cloud

Operazione	Description	Competenze richieste
Visualizza i log dell'applicazione di esempio in CloudWatch Logs.	Visualizza i log associati alla funzione Lambda `<sample-app-function-name>` denominata CloudWatch in Logs. Per istruzioni, consulta Invio dei log delle funzioni Lambda ai CloudWatch registri nella documentazione di Lambda. È possibile visualizzare la politica basata sull'ambito dei tenant generata da TVM in questi log. Questa politica con ambito tenant fornisce le autorizzazioni per l'applicazione di esempio ad Amazon S3,, e PutObjectGetObject DeleteObjectListBucket APIs, ma solo per il prefisso dell'oggetto a cui è associato . `<sample-tenant-name>` Nelle chiamate successive dell'applicazione di esempio, in caso di modifiche`<sample-tenant-name>` , TVM aggiorna la policy di ambito in modo che corrisponda al tenant fornito nel payload di chiamata. Questa policy generata dinamicamente mostra come è possibile mantenere l'accesso con ambito tenant con una TVM nelle applicazioni SaaS.	Amministratore cloud

Operazione	Description	Competenze richieste
	La funzionalità TVM è fornita in un livello Lambda in modo che possa essere collegata ad altre funzioni Lambda utilizzate da un'applicazione senza dover replicare il codice. Per un'illustrazione della policy generata dinamicamente, consultate la sezione Informazioni aggiuntive.	

Risorse correlate

- Isolamento dei tenant con policy IAM generate dinamicamente (post sul blog)
- Applicazione di politiche di isolamento generate dinamicamente in ambienti SaaS (post sul blog)
- SaaS attivo AWS

Informazioni aggiuntive

Il registro seguente mostra la politica generata dinamicamente dal codice TVM secondo questo schema. In questa schermata, il `<sample-app-bucket-name>` è `DOC-EXAMPLE-BUCKET` e il è. `<sample-tenant-name>` `test-tenant-1` Le credenziali STS restituite da questa policy con ambito non sono in grado di eseguire alcuna azione sugli oggetti nel bucket S3 ad eccezione degli oggetti associati al prefisso object key. `test-tenant-1`

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Implementa il modello di saga serverless utilizzando AWS Step Functions

Tabby Ward, Joe Kern e Rohan Mehta, Amazon Web Services

Riepilogo

In un'architettura di microservizi, l'obiettivo principale è creare componenti disaccoppiati e indipendenti per promuovere agilità, flessibilità e tempi di commercializzazione più rapidi per le applicazioni. Come risultato del disaccoppiamento, ogni componente dei microservizi ha il proprio livello di persistenza dei dati. In un'architettura distribuita, le transazioni commerciali possono estendersi su più microservizi. Poiché questi microservizi non possono utilizzare una singola transazione ACID (atomicità, coerenza, isolamento, durabilità), è possibile che si ottengano transazioni parziali. In questo caso, è necessaria una certa logica di controllo per annullare le transazioni che sono già state elaborate. Il modello a saga distribuito viene in genere utilizzato per questo scopo.

Il modello a saga è un modello di gestione degli errori che aiuta a stabilire la coerenza nelle applicazioni distribuite e coordina le transazioni tra più microservizi per mantenere la coerenza dei dati. Quando si utilizza il modello saga, ogni servizio che esegue una transazione pubblica un evento che attiva i servizi successivi per eseguire la transazione successiva nella catena. Questo continua fino al completamento dell'ultima transazione della catena. Se una transazione commerciale fallisce, saga orchestra una serie di transazioni di compensazione che annullano le modifiche apportate dalle transazioni precedenti.

Questo modello dimostra come automatizzare la configurazione e la distribuzione di un'applicazione di esempio (che gestisce le prenotazioni di viaggi) con tecnologie serverless come AWS Step Functions, AWS Lambda e Amazon DynamoDB. L'applicazione di esempio utilizza anche Amazon API Gateway e Amazon Simple Notification Service (Amazon SNS) per implementare un coordinatore dell'esecuzione di saga. Il pattern può essere distribuito con un framework Infrastructure as code (IaC) come AWS Cloud Development Kit (AWS CDK), AWS Serverless Application Model (AWS SAM) o Terraform.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Autorizzazioni per creare uno CloudFormation stack AWS. Per ulteriori informazioni, consulta Controllare l'accesso nella CloudFormation documentazione.

- Framework IaC di tua scelta (AWS CDK, AWS SAM o Terraform) configurato con il tuo account AWS in modo da poter utilizzare la CLI del framework per distribuire l'applicazione.
- NodeJS, utilizzato per creare l'applicazione ed eseguirla localmente.
- Un editor di codice a tua scelta (come Visual Studio Code, Sublime o Atom).

Versioni del prodotto

- NodeJS versione 14
- CDK AWS versione 2.37.1
- AWS SAM versione 1.71.0
- Terraform versione 1.3.7

Limitazioni

L'event sourcing è un modo naturale per implementare il modello di orchestrazione della saga in un'architettura di microservizi in cui tutti i componenti sono liberamente accoppiati e non hanno una conoscenza diretta l'uno dell'altro. Se la transazione prevede un numero limitato di passaggi (da tre a cinque), il modello a saga potrebbe essere la soluzione ideale. Tuttavia, la complessità aumenta con il numero di microservizi e il numero di passaggi.

Il test e il debug possono diventare difficili quando si utilizza questo design, poiché è necessario che tutti i servizi siano in esecuzione per simulare il modello di transazione.

Architecture

Architettura di Target

L'architettura proposta utilizza AWS Step Functions per creare uno schema a saga per prenotare voli, prenotare auto a noleggio ed elaborare i pagamenti per le vacanze.

Il seguente diagramma del flusso di lavoro illustra il flusso tipico del sistema di prenotazione viaggi. Il flusso di lavoro consiste nella prenotazione di viaggi aerei (» ReserveFlight «), nella prenotazione di un'auto (» ReserveCarRental «), nell'elaborazione dei pagamenti (» ProcessPayment «), nella conferma delle prenotazioni dei voli (» ConfirmFlight «) e nella conferma del noleggio auto (» ConfirmCarRental «) seguite da una notifica di avvenuto completamento di questi passaggi. Tuttavia, se il sistema riscontra errori nell'esecuzione di una di queste transazioni, inizia a fallire all'indietro. Ad esempio, un errore nell'elaborazione dei pagamenti (» ProcessPayment «) attiva un

rimborso (» RefundPayment «), che quindi attiva la cancellazione dell'auto a noleggio e del volo (» CancelRentalReservation "e" CancelFlightReservation «), che termina l'intera transazione con un messaggio di errore.

Questo modello implementa funzioni Lambda separate per ogni attività evidenziata nel diagramma, oltre a tre tabelle DynamoDB per voli, autonoleggi e pagamenti. Ogni funzione Lambda crea, aggiorna o elimina le righe nelle rispettive tabelle DynamoDB, a seconda che una transazione sia confermata o ripristinata. Il modello utilizza Amazon SNS per inviare messaggi di testo (SMS) agli abbonati, notificandoli delle transazioni non riuscite o riuscite.

Automazione e scalabilità

È possibile creare la configurazione per questa architettura utilizzando uno dei framework IaC. Usa uno dei seguenti link per il tuo IAc preferito.

- Implementa con AWS CDK
- Implementa con AWS SAM
- Implementa con Terraform

Tools (Strumenti)

Servizi AWS

- AWS Step Functions è un servizio di orchestrazione serverless che consente di combinare funzioni AWS Lambda e altri servizi AWS per creare applicazioni aziendali critiche. Tramite la console grafica Step Functions, puoi vedere il flusso di lavoro dell'applicazione come una serie di passaggi guidati dagli eventi.
- Amazon DynamoDB è un servizio di database NoSQL completamente gestito che offre prestazioni veloci e prevedibili con una scalabilità perfetta. Puoi utilizzare DynamoDB per creare una tabella di database in grado di archiviare e recuperare qualunque quantità di dati e soddisfare qualsiasi livello di traffico di richiesto.
- AWS Lambda è un servizio di elaborazione che consente di eseguire codice senza effettuare il provisioning o la gestione di server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo.
- Amazon API Gateway è un servizio AWS per la creazione, la pubblicazione, la manutenzione, il monitoraggio e la protezione di REST, HTTP e WebSocket APIs su qualsiasi scala.

- Amazon Simple Notification Service (Amazon SNS) è un servizio gestito che fornisce il recapito dei messaggi dagli editori agli abbonati.
- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software per definire le risorse delle applicazioni cloud utilizzando linguaggi di programmazione familiari come Python TypeScript JavaScript, Java e C#/.Net.
- AWS Serverless Application Model (AWS SAM) Serverless Application Model (AWS) Serverless Application Model (AWS SAM) è un framework open source per la creazione di applicazioni serverless. Fornisce una sintassi abbreviata per esprimere funzioni, database e mappature delle sorgenti degli APIs eventi.

Codice

Il codice per un'applicazione di esempio che dimostra il modello saga, incluso il modello IaC (AWS CDK, AWS SAM o Terraform), le funzioni Lambda e le tabelle DynamoDB, è disponibile nei seguenti link. Segui le istruzioni del primo capitolo epico per installarli.

- Implementa con AWS CDK
- Implementa con AWS SAM
- Implementa con Terraform

Epiche

Installa pacchetti, compila e crea

Operazione	Description	Competenze richieste
Installa i pacchetti NPM.	Crea una nuova directory, accedi a quella directory in un terminale e clona il GitHub repository di tua scelta dalla sezione Codice precedente di questo schema. Nella cartella principale che contiene il `package.json` file, esegui il seguente	Sviluppatore, architetto cloud

Operazione	Description	Competenze richieste
	comando per scaricare e installare tutti i pacchetti Node Package Manager (NPM): `npm install`	
Compila script.	Nella cartella principale, esegui il seguente comando per indicare al TypeScript transpiler di creare tutti i file necessari: JavaScript `npm run build`	Sviluppatore, architetto cloud
Controlla le modifiche e ricompila.	Nella cartella principale, esegui il seguente comando in una finestra di terminale separata per controllare le modifiche al codice e compila il codice quando rileva una modifica: `npm run watch`	Sviluppatore, architetto cloud
Esegui test unitari (solo AWS CDK).	Se utilizzi il CDK AWS, nella cartella principale, esegui il seguente comando per eseguire gli unit test di Jest: `npm run test`	Sviluppatore, architetto del cloud

Distribuisci risorse sull'account AWS di destinazione

Operazione	Description	Competenze richieste

Distribuisci lo stack dimostrativo su AWS.	**Important**
	L'applicazione è indipendente dalla regione AWS. Se utilizzi un profilo, devi dichiarare la regione in modo esplicito nel profilo AWS Command Line Interface (AWS CLI) o tramite le variabili di ambiente AWS CLI.
	
	Nella cartella principale, esegui il comando seguente per creare un assembly di distribuzione e distribuirlo nell'account e nella regione AWS predefiniti.
	
	CDK AWS:
	
	`cdk bootstrap`
	`cdk deploy`
	
	AWS È:
	
	`sam build`
	`sam deploy --guided`
	
	Terraforma:
		Sviluppatore, architetto cloud

Operazione	Description	Competenze richieste

	`terraform init`
	`terraform apply`
	
	Il completamento di questo passaggio potrebbe richiedere alcuni minuti. Questo comando utilizza le credenziali predefinite configurate per l'AWS CLI.
	
	Nota l'URL dell'API Gateway che viene visualizzato sulla console al termine della distribuzione. Avrai bisogno di queste informazioni per testare il flusso di esecuzione della saga.	

Operazione	Description	Competenze richieste

Confronta lo stack distribuito con lo stato attuale.	Nella cartella principale, esegui il comando seguente per confrontare lo stack distribuito con lo stato corrente dopo aver apportato modifiche al codice sorgente:
	
	CDK AWS:
	
	`cdk diff`
	
	AWS È:
	
	`sam deploy`
	
	Terraforma:
	
	`terraform plan`
		Sviluppatore, architetto cloud

Testa il flusso di esecuzione

Operazione	Description	Competenze richieste
Prova il flusso di esecuzione della saga.	Passa all'URL dell'API Gateway che hai annotato nel passaggio precedent e, quando hai distribuito lo stack. Questo URL attiva l'avvio della macchina a stati. Per ulteriori informazi oni su come manipolare il flusso della macchina a stati passando diversi parametri	Sviluppatore, architetto cloud

Operazione	Description	Competenze richieste
	URL, consulta la sezione Informazioni aggiuntive. Per visualizzare i risultati , accedi alla Console di gestione AWS e vai alla console Step Functions. Qui puoi vedere ogni fase della saga state machine. Puoi anche visualizzare la tabella DynamoDB per vedere i record inseriti, aggiornati o eliminati. Se aggiorni spesso la schermata, puoi vedere lo stato della transazione cambiare da a. `pending` `confirmed` Puoi iscriverti all'argomento SNS aggiornando il codice nel `stateMachine.ts` file con il tuo numero di cellulare per ricevere messaggi SMS in caso di prenotazioni riuscite o non riuscite. Per ulteriori informazioni, consulta Amazon SNS nella sezione Informazioni aggiuntive.	

Eliminazione

Operazione	Description	Competenze richieste
Eliminare le risorse.	Per ripulire le risorse distribuite per questa applicazione,	Sviluppatore di app, architetto cloud

Operazione	Description	Competenze richieste

	è possibile utilizzare uno dei seguenti comandi.
	
	CDK AWS:
	
	`cdk destroy`
	
	AWS È:
	
	`sam delete`
	
	Terraforma:
	
	`terraform destroy`	

Risorse correlate

Documenti tecnici

- Implementazione di microservizi su AWS
- Lente applicativa serverless

Documentazione del servizio AWS

- Guida introduttiva alla CDK AWS
- Guida introduttiva ad AWS SAM
- AWS Step Functions
- Amazon DynamoDB
- AWS Lambda
- Gateway Amazon API
- Amazon SNS

Tutorial

- Workshop pratici per l'informatica serverless

Informazioni aggiuntive

Codice

A scopo di test, questo modello implementa API Gateway e una funzione Lambda di test che attiva la macchina a stati Step Functions. Con Step Functions, puoi controllare la funzionalità del sistema di prenotazione viaggi passando un `run_type` parametro per simulare gli errori in "ReserveFlightReserveCarRental,» "ProcessPayment,»,» e "ConfirmFlightConfirmCarRental.»

La funzione `saga` Lambda (`sagaLambda.ts`) riceve l'input dai parametri di query nell'URL API Gateway, crea il seguente oggetto JSON e lo passa a Step Functions per l'esecuzione:

```
let input = {
"trip_id": tripID, //  value taken from query parameter, default is AWS request ID
"depart_city": "Detroit",
"depart_time": "2021-07-07T06:00:00.000Z",
"arrive_city": "Frankfurt",
"arrive_time": "2021-07-09T08:00:00.000Z",
"rental": "BMW",
"rental_from": "2021-07-09T00:00:00.000Z",
"rental_to": "2021-07-17T00:00:00.000Z",
"run_type": runType // value taken from query parameter, default is "success"
};
```

È possibile sperimentare diversi flussi della macchina a stati Step Functions passando i seguenti parametri URL:

- Esecuzione riuscita – https://{api gateway url}
- Reserve Flight Fail – https://{api gateway url}? RunType= failFlightsReservation
- Conferma il fallimento del volo – https://{api gateway url}? RunType= failFlightsConfirmation
- Reserve Car Rental Failure – https://{api gateway url}? RunType= Prenotazione failCarRental
- Conferma il fallimento del noleggio auto – https://{api gateway url}? RunType= failCarRental Conferma
- Processo di pagamento non riuscito – https://{api gateway url}? RunType=FailPayment
- Passa un Trip ID – https://{api gateway url}? tripId= {per impostazione predefinita, l'ID del viaggio sarà l'ID della richiesta AWS}

modelli IaC

Gli archivi collegati includono modelli IaC che è possibile utilizzare per creare l'intera applicazione di prenotazione viaggi di esempio.

- Implementa con AWS CDK
- Implementa con AWS SAM
- Implementa con Terraform

Tabelle DynamoDB

Ecco i modelli di dati per le tabelle dei voli, degli autonoleggi e dei pagamenti.

```
Flight Data Model:
 var params = {
      TableName: process.env.TABLE_NAME,
      Item: {
        'pk' : {S: event.trip_id},
        'sk' : {S: flightReservationID},
        'trip_id' : {S: event.trip_id},
        'id': {S: flightReservationID},
        'depart_city' : {S: event.depart_city},
        'depart_time': {S: event.depart_time},
        'arrive_city': {S: event.arrive_city},
        'arrive_time': {S: event.arrive_time},
        'transaction_status': {S: 'pending'}
      }
    };

Car Rental Data Model:
var params = {
      TableName: process.env.TABLE_NAME,
      Item: {
        'pk' : {S: event.trip_id},
        'sk' : {S: carRentalReservationID},
        'trip_id' : {S: event.trip_id},
        'id': {S: carRentalReservationID},
        'rental': {S: event.rental},
        'rental_from': {S: event.rental_from},
        'rental_to': {S: event.rental_to},
        'transaction_status': {S: 'pending'}
      }
```

```
    };

Payment Data Model:
var params = {
      TableName: process.env.TABLE_NAME,
      Item: {
        'pk' : {S: event.trip_id},
        'sk' : {S: paymentID},
        'trip_id' : {S: event.trip_id},
        'id': {S: paymentID},
        'amount': {S: "750.00"}, // hard coded for simplicity as implementing any
 monetary transaction functionality is beyond the scope of this pattern
        'currency': {S: "USD"},
        'transaction_status': {S: "confirmed"}
      }
    };
```

Funzioni Lambda

Verranno create le seguenti funzioni per supportare il flusso e l'esecuzione della macchina a stati in Step Functions:

- Reserve Flights: inserisce un record nella tabella DynamoDB Flights con `transaction_status` un `pending` di, per prenotare un volo.
- Confirm Flight: aggiorna il record nella tabella DynamoDB Flights, da `transaction_status` impostare su, per confermare `confirmed` il volo.
- Annulla prenotazione voli: elimina il record dalla tabella DynamoDB Flights, per annullare il volo in sospeso.
- Reserve Car Rentals: inserisce un record nella tabella CarRentals DynamoDB con `transaction_status` un `pending` di, per prenotare un noleggio auto.
- Conferma noleggio auto: aggiorna il record nella tabella CarRentals DynamoDB, `transaction_status` impostandolo su, per confermare `confirmed` il noleggio auto.
- Annulla prenotazione noleggio auto: elimina il record dalla tabella CarRentals DynamoDB, per annullare il noleggio auto in sospeso.
- Elabora pagamento: inserisce un record nella tabella Pagamenti di DynamoDB per il pagamento.
- Annulla pagamento: elimina il record del pagamento dalla tabella DynamoDB Payments.

Amazon SNS

L'applicazione di esempio crea il seguente argomento e sottoscrizione per l'invio di messaggi SMS e la notifica al cliente in caso di prenotazioni riuscite o non riuscite. Se desideri ricevere messaggi di testo durante il test dell'applicazione di esempio, aggiorna l'abbonamento SMS con il tuo numero di telefono valido nel file di definizione della macchina a stati.

Frammento di CDK AWS (aggiungi il numero di telefono nella seconda riga del codice seguente):

```
const topic = new  sns.Topic(this, 'Topic');
topic.addSubscription(new subscriptions.SmsSubscription('+11111111111'));
const snsNotificationFailure = new tasks.SnsPublish(this ,'SendingSMSFailure', {
topic:topic,
integrationPattern: sfn.IntegrationPattern.REQUEST_RESPONSE,
message: sfn.TaskInput.fromText('Your Travel Reservation Failed'),
});

const snsNotificationSuccess = new tasks.SnsPublish(this ,'SendingSMSSuccess', {
topic:topic,
integrationPattern: sfn.IntegrationPattern.REQUEST_RESPONSE,
message: sfn.TaskInput.fromText('Your Travel Reservation is Successful'),
});
```

Snippet AWS SAM (sostituisci le +1111111111 stringhe con il tuo numero di telefono valido):

```
  StateMachineTopic11111111111:
    Type: 'AWS::SNS::Subscription'
    Properties:
      Protocol: sms
      TopicArn:
        Ref: StateMachineTopic
      Endpoint: '+11111111111'
    Metadata:
      'aws:sam:path': SamServerlessSagaStack/StateMachine/Topic/+11111111111/Resource
```

Snippet Terraform (sostituisci la +111111111 stringa con il tuo numero di telefono valido):

```
resource "aws_sns_topic_subscription" "sms-target" {
  topic_arn = aws_sns_topic.topic.arn
  protocol  = "sms"
  endpoint  = "+11111111111"
}
```

Prenotazioni riuscite

Il seguente flusso illustra una prenotazione andata a buon fine con "ReserveFlight,»ReserveCarRental,» e "ProcessPayment" seguiti da "ConfirmFlight" e "»ConfirmCarRental. Il cliente viene informato dell'avvenuta prenotazione tramite messaggi SMS che vengono inviati all'abbonato dell'argomento SNS.

Prenotazioni non riuscite

Questo flusso è un esempio di fallimento dello schema della saga. Se, dopo aver prenotato voli e auto a noleggio, "ProcessPayment" fallisce, i passaggi vengono annullati in ordine inverso. Le prenotazioni vengono rilasciate e il cliente viene informato dell'errore tramite messaggi SMS inviati all'abbonato dell'argomento SNS.

Gestisci le applicazioni container locali configurando Amazon ECS Anywhere con AWS CDK

Dott. Rahul Sharad Gaikwad, Amazon Web Services

Riepilogo

Amazon ECS Anywhere è un'estensione di Amazon Elastic Container Service (Amazon ECS). Puoi usare ECS Anywhere per distribuire attività native di Amazon ECS in un ambiente locale o gestito dal cliente. Questa funzionalità aiuta a ridurre i costi e mitigare l'orchestrazione e le operazioni complesse dei container locali. Puoi utilizzare ECS Anywhere per distribuire ed eseguire applicazioni container in ambienti sia locali che cloud. Elimina la necessità per il team di apprendere più domini e set di competenze o di gestire software complessi da solo.

Questo modello illustra i passaggi per configurare ECS Anywhere utilizzando gli stack AWS Cloud Development Kit (AWS CDK).

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- AWS Command Line Interface (AWS CLI), installata e configurata. (Vedi Installazione, aggiornamento e disinstallazione dell'interfaccia a riga di comando di AWS nella documentazione dell'interfaccia a riga di comando di AWS.)
- AWS CDK Toolkit, installato e configurato. (Consulta AWS CDK Toolkit nella documentazione di AWS CDK e segui le istruzioni per installare la versione 2 a livello globale).
- Node package manager (npm), installato e configurato per AWS CDK in. TypeScript (Vedi Download e installazione di Node.js e npm nella documentazione di npm.)

Limitazioni

- Per limitazioni e considerazioni, consulta Istanze esterne (Amazon ECS Anywhere) nella documentazione di Amazon ECS.

Versioni del prodotto

- AWS CDK Toolkit versione 2

- npm versione 7.20.3 o successiva
- Node.js versione 16.6.1 o successiva

Architecture

Stack tecnologico Target

- AWS CloudFormation
- AWS CDK
- Amazon ECS Anywhere
- AWS Identity and Access Management (IAM)

Architettura Target

Il diagramma seguente illustra un'architettura di sistema di alto livello di configurazione di ECS Anywhere che utilizza AWS CDK con TypeScript, come implementato da questo modello.

1. Quando distribuisci lo stack CDK AWS, viene creato uno CloudFormation stack su AWS.
2. Lo CloudFormation stack fornisce un cluster Amazon ECS e le relative risorse AWS.
3. Per registrare un'istanza esterna con un cluster Amazon ECS, devi installare AWS Systems Manager Agent (SSM Agent) sulla tua macchina virtuale (VM) e registrare la macchina virtuale come istanza gestita da AWS Systems Manager.
4. È inoltre necessario installare l'agente contenitore Amazon ECS e Docker sulla macchina virtuale per registrarla come istanza esterna nel cluster Amazon ECS.
5. Quando l'istanza esterna è registrata e configurata con il cluster Amazon ECS, può eseguire più contenitori sulla macchina virtuale, che è registrata come istanza esterna.

Automazione e scalabilità

Il GitHub repository fornito con questo modello utilizza AWS CDK come strumento Infrastructure as Code (IaC) per creare la configurazione per questa architettura. AWS CDK ti aiuta a orchestrare le risorse e configurare ECS Anywhere.

Tools (Strumenti)

- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software che aiuta a definire e fornire l'infrastruttura cloud AWS in codice.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.

Codice

Il codice sorgente per questo pattern è disponibile nel GitHub repository Amazon ECS Anywhere CDK Samples. Per clonare e utilizzare il repository, segui le istruzioni nella sezione successiva.

Epiche

Verifica la configurazione di AWS CDK

Operazione	Description	Competenze richieste
Verifica la versione di AWS CDK.	Verifica la versione di AWS CDK Toolkit eseguendo il seguente comando: `cdk --version` Questo modello richiede la versione 2 di AWS CDK. Se disponi di una versione precedente di AWS CDK, segui le istruzioni nella documentazione di AWS CDK per aggiornarla.	DevOps ingegnere
Configura le credenziali AWS.	Per configurare le credenziali, esegui il `aws configure` comando e segui le istruzioni: `$aws configure`	DevOps ingegnere

Operazione	Description	Competenze richieste

	`AWS Access Key ID [None]: <your-access-key-ID>`
	`AWS Secret Access Key [None]: <your-secret-access-key>`
	`Default region name [None]: <your-Region-name>`
	`Default output format [None]:`	

Esegui il bootstrap dell'ambiente AWS CDK

Operazione	Description	Competenze richieste
Clona il repository di codice AWS CDK.	Clona il repository di GitHub codice per questo pattern usando il comando: `git clone https://github.com/aws-samples/amazon-ecs-anywhere-cdk-samples.git`	DevOps ingegnere
Avvia l'ambiente.	Per distribuire il CloudFormation modello AWS nell'account e nella regione AWS che desideri utilizzare, esegui il seguente comando: `cdk bootstrap <account-number>/<Region>`	DevOps ingegnere

Operazione	Description	Competenze richieste
	Per ulteriori informazioni, consulta Bootstrapping nella documentazione di AWS CDK.	

Crea e distribuisci il progetto

Operazione	Description	Competenze richieste

Installa le dipendenze dei pacchetti e compila i file TypeScript .	Installa le dipendenze del pacchetto e compila TypeScript i file eseguendo i seguenti comandi:
	`$cd amazon-ecs-anywhere-cdk-samples`
	`$npm install`
	`$npm fund`
	Questi comandi installano tutti i pacchetti dal repository di esempio.
	Important
	Se ricevi errori sui pacchetti mancanti, usa uno dei seguenti comandi:
	`$npm ci`
	oppure
		DevOps ingegnere

Operazione	Description	Competenze richieste
	`$npm install -g @aws-cdk/<package_name>` Per ulteriori informazioni, consulta npm ci e npm install nella documentazione di npm.	
Compilare il progetto.	Per creare il codice del progetto, esegui il comando: `npm run build` Per ulteriori informazioni sulla creazione e la distribuzione del progetto, consulta La tua prima app AWS CDK nella documentazione di AWS CDK.	DevOps ingegnere
Distribuisci il progetto	Per distribuire il codice del progetto, esegui il comando: `cdk deploy`	DevOps ingegnere
Verifica la creazione e l'output dello stack.	Apri la CloudFormation console AWS all'indirizzo https://console.aws.amazon.com/cloudformation e scegli lo stack. `EcsAnywhereStack` La scheda Outputs mostra i comandi da eseguire sulla macchina virtuale esterna.	DevOps ingegnere

Configura una macchina locale

Operazione	Description	Competenze richieste
Configura la tua VM usando Vagrant.	A scopo dimostrativo, puoi usare HashiCorp Vagrant per creare una macchina virtuale. Vagrant è un'utilità open source per la creazione e la manutenzione di ambienti di sviluppo software virtuali portatili. Crea una VM Vagrant eseguendo il `vagrant up` comando dalla directory principale in cui è posiziona to Vagrantfile. Per ulteriori informazioni, consulta la documentazione di Vagrant.	DevOps ingegnere
Registra la tua macchina virtuale come istanza esterna.	1. Accedi alla VM Vagrant utilizzando il comando. `vagrant ssh` Per ulteriori informazioni, consulta la documentazione di Vagrant. 2. Crea un codice di attivazio ne e un ID che puoi utilizzare per registrare la tua macchina virtuale con AWS Systems Manager e attivare l'istanza esterna. L'output di questo comando include `Activatio nId` e `ActivationCode` valori: `aws ssm create-ac tivation --iam-role EcsAnywhereInstanc`	DevOps ingegnere

Operazione	Description	Competenze richieste

	`eRole | tee ssm-activ ation.json`
	
	3. Esporta l'ID di attivazione e i valori del codice:
	
	`export ACTIVATIO N_ID=<activation-ID> export ACTIVATIO N_CODE=<activation- code>`
	
	4. Scarica lo script di installaz ione sul server o sulla macchina virtuale locale:
	
	`curl -o "ecs-anywhere- install.sh" "https:// amazon-ecs-agent.s 3.amazonaws.com/ec s-anywhere-install -latest.sh" && sudo chmod +x ecs-anywhere- install.sh`
	
	5. Esegui lo script di installaz ione sul server o sulla macchina virtuale locale:
	
	`sudo ./ecs-anywhere-ins tall.sh \ --cluster test-ecs- anywhere \ --activation-id $ACTIVATION_ID \ --activation-code $ACTIVATION_CODE \ --region <Region>`	

Operazione	Description	Competenze richieste
	Per ulteriori informazioni sulla configurazione e la registrazione di una macchina virtuale, consulta Registrazione di un'istanza esterna in un cluster nella documentazione di Amazon ECS.	

Verifica lo stato di ECS Anywhere e della macchina virtuale esterna.	Per verificare se la tua casella virtuale è connessa al piano di controllo di Amazon ECS e se è in funzione, usa i seguenti comandi:
	`aws ssm describe-instance-information`
	`aws ecs list-container-instances --cluster $CLUSTER_NAME`
		DevOps ingegnere

Eliminazione

Operazione	Description	Competenze richieste
Pulisci ed elimina le risorse.	Dopo aver seguito questo schema, dovresti rimuovere le risorse che hai creato per evitare di incorrere in ulteriori addebiti. Per pulire, esegui il comando: `cdk destroy`	DevOps ingegnere

Risorse correlate

- Documentazione di Amazon ECS Anywhere
- Demo di Amazon ECS Anywhere
- Esempi di workshop su Amazon ECS Anywhere

Modernizza le applicazioni ASP.NET Web Forms su AWS

Vijai Anand Ramalingam e Sreelaxmi Pai, Amazon Web Services

Riepilogo

Questo modello descrive i passaggi per modernizzare un'applicazione ASP.NET Web Forms legacy e monolitica portandola su ASP.NET Core su AWS.

Il trasferimento delle applicazioni ASP.NET Web Forms su ASP.NET Core consente di sfruttare le prestazioni, i risparmi sui costi e il robusto ecosistema di Linux. Tuttavia, può essere un notevole sforzo manuale. In questo modello, l'applicazione legacy viene modernizzata in modo incrementale utilizzando un approccio graduale e quindi containerizzata nel cloud AWS.

Prendi in considerazione un'applicazione monolitica legacy per un carrello della spesa. Supponiamo che sia stata creata come applicazione ASP.NET Web Forms e sia costituita da pagine.aspx con un file code-behind (). `aspx.cs` Il processo di modernizzazione consiste nei seguenti passaggi:

1. Suddividi il monolite in microservizi utilizzando i modelli di scomposizione appropriati. Per ulteriori informazioni, consulta la guida Decomposizione dei monoliti in microservizi sul sito Web AWS Prescriptive Guidance.
2. Trasferisci la tua applicazione ASP.NET Web Forms (.NET Framework) legacy su ASP.NET Core in .NET 5 o versione successiva. In questo modello, si utilizza Porting Assistant for .NET per analizzare l'applicazione ASP.NET Web Forms e identificare le incompatibilità con ASP.NET Core. Ciò riduce lo sforzo di portabilità manuale.
3. Risviluppa il livello dell'interfaccia utente di Web Forms utilizzando React. Questo modello non copre la riqualificazione dell'interfaccia utente. Per istruzioni, consulta Creare una nuova app React nella documentazione di React.
4. Risviluppa il file code-behind di Web Forms (interfaccia aziendale) come API web ASP.NET Core. Questo modello utilizza i NDepend report per aiutare a identificare i file e le dipendenze necessari.
5. Aggiorna shared/common progetti, come Business Logic e Data Access, nell'applicazione precedente a .NET 5 o versione successiva utilizzando Porting Assistant for .NET.
6. Aggiungi i servizi AWS per completare la tua applicazione. Ad esempio, puoi utilizzare Amazon CloudWatch Logs per monitorare, archiviare e accedere ai log delle tue applicazioni e AWS Systems Manager per archiviare le impostazioni delle applicazioni.
7. Containerizza l'applicazione ASP.NET Core modernizzata. Questo modello crea un file Docker destinato a Linux in Visual Studio e utilizza Docker Desktop per testarlo localmente. Questo

passaggio presuppone che l'applicazione legacy sia già in esecuzione su un'istanza Windows locale o Amazon Elastic Compute Cloud (Amazon EC2). Per ulteriori informazioni, consulta lo schema Esegui un contenitore Docker dell'API Web ASP.NET Core su un'istanza Amazon EC2 Linux.

8. Distribuisci l'applicazione core ASP.NET modernizzata su Amazon Elastic Container Service (Amazon ECS). Questo modello non copre la fase di implementazione. Per istruzioni, consulta Amazon ECS Workshop.

> **Note**
>
> Questo modello non copre lo sviluppo dell'interfaccia utente, la modernizzazione del database o le fasi di implementazione dei container.

Prerequisiti e limitazioni

Prerequisiti

- Visual Studio o Visual Studio Code, scaricato e installato.
- Accesso a un account AWS utilizzando la Console di gestione AWS e l'AWS Command Line Interface (AWS CLI) versione 2. (Consulta le istruzioni per configurare l'interfaccia a riga di comando di AWS).
- AWS Toolkit for Visual Studio (vedi le istruzioni di configurazione).
- Docker Desktop, scaricato e installato.
- .NET SDK, scaricato e installato.
- NDepend strumento, scaricato e installato. Per installare l' NDepend estensione per Visual Studio, esegui `NDepend.VisualStudioExtension.Installer` (vedi istruzioni). Puoi selezionare Visual Studio 2019 o 2022, a seconda delle tue esigenze.
- Porting Assistant per .NET, scaricato e installato.

Architecture

Modernizzazione dell'applicazione del carrello degli acquisti

Il diagramma seguente illustra il processo di modernizzazione di un'applicazione esistente per il carrello degli acquisti ASP.NET.

Architettura Target

Il diagramma seguente illustra l'architettura dell'applicazione modernizzata per il carrello degli acquisti su AWS. APIs I Web ASP.NET Core vengono distribuiti in un cluster Amazon ECS. I servizi di registrazione e configurazione sono forniti da Amazon CloudWatch Logs e AWS Systems Manager.

Tools (Strumenti)

Servizi AWS

- Amazon ECS — Amazon Elastic Container Service (Amazon ECS) è un servizio di gestione dei container veloce e altamente scalabile per l'esecuzione, l'arresto e la gestione dei container su un cluster. Puoi eseguire attività e servizi su un'infrastruttura serverless gestita da AWS Fargate. In alternativa, per un maggiore controllo sulla tua infrastruttura, puoi eseguire attività e servizi su un cluster di EC2 istanze da te gestito.
- Amazon CloudWatch Logs: Amazon CloudWatch Logs centralizza i log di tutti i sistemi, le applicazioni e i servizi AWS che utilizzi. Puoi visualizzare e monitorare i log, cercarli per codici o modelli di errore specifici, filtrarli in base a campi specifici o archiviarli in modo sicuro per analisi future.
- AWS Systems Manager – AWS Systems Manager è un servizio AWS che puoi usare per visualizzare e controllare la tua infrastruttura su AWS. Utilizzando la console Systems Manager, puoi visualizzare i dati operativi da più servizi AWS e automatizzare le attività operative tra le tue risorse AWS. Systems Manager ti aiuta a mantenere la sicurezza e la conformità scansionando le istanze gestite e segnalando (o adottando azioni correttive) su eventuali violazioni delle policy rilevate.

Strumenti

- Visual Studio o Visual Studio Code: strumenti per la creazione di applicazioni.NET APIs, Web e altri programmi.
- AWS Toolkit for Visual Studio: un'estensione per Visual Studio che aiuta a sviluppare, eseguire il debug e distribuire applicazioni.NET che utilizzano i servizi AWS.
- Docker Desktop: uno strumento che semplifica la creazione e la distribuzione di applicazioni containerizzate.

- NDepend— Un analizzatore che monitora il codice.NET per individuare dipendenze, problemi di qualità e modifiche al codice.
- Porting Assistant for .NET: strumento di analisi che analizza il codice.NET per identificare le incompatibilità con .NET Core e stimare lo sforzo di migrazione.

Epiche

Trasferisci la tua applicazione precedente su.NET 5 o versione successiva

Operazione	Description	Competenze richieste
Aggiorna la tua applicazione.NET Framework legacy a.NET 5.	È possibile utilizzare Porting Assistant for .NET per convertire l'applicazione ASP.NET Web Forms legacy in .NET 5 o versione successiva. Segui le istruzioni contenute nella documentazione di Porting Assistant for .NET.	Sviluppatore di app
Genera NDepend report.	Quando modernizzi l'applicazione ASP.NET Web Forms scomponendola in microservizi, potresti non aver bisogno di tutti i file.cs dell'applicazione precedente. È possibile utilizzarlo NDepend per generare un report per qualsiasi file code-behind (.cs), per ottenere tutti i chiamanti e i chiamanti. Questo rapporto consente di identificare e utilizzare solo i file richiesti nei microservizi. Dopo l'installazione NDepend (vedi la sezione Prerequisiti),	Sviluppatore di app

Operazione	Description	Competenze richieste
	apri la soluzione (file.sln) per l'applicazione precedente in Visual Studio e segui questi passaggi: 1. Crea l'applicazione legacy in Visual Studio. 2. Nella barra dei menu di Visual Studio NDepend, scegli Allega nuovo NDepend progetto alla soluzione VS corrente. 3. Scegli Analizza gli assiemi.NET. 4. Una volta completata l'analisi, accedi al progetto in Solution Explorer. Fate clic con il pulsante destro del mouse su qualsiasi file code-behind (ad esempio`listproducts.aspx.cs`) per cui desiderate generare il report, quindi scegliete Mostra sul grafico delle dipendenze. 5. Nella barra di navigazione, scegli Chiamanti e chiamanti, quindi scegli Modifica query di codice. 6. Nel riquadro di modifica delle interrogazioni e delle regole, scegli la freccia di	

Operazione	Description	Competenze richieste
	download, quindi scegli Esporta in Excel. Questo processo genera un report per il file code-behind che elenca tutti i chiamanti e i chiamanti. Per ulteriori informazioni sul grafico delle dipendenze, consulta la documentazione. NDepend	

Operazione	Description	Competenze richieste
Crea una nuova soluzione .NET 5.	Per creare una nuova struttura.NET 5 (o versione successiva) per il tuo sito Web ASP.NET Core modernizzato: APIs 1. Apri Visual Studio. 2. Crea una nuova soluzione vuota. 3. Crea nuovi progetti destinati a .NET 5 (o versioni successive), in base alla tua applicazione legacy. Per esempi di progetti nuovi e precedenti per un'applicazione per il carrello degli acquisti, consulta la sezione Informazioni aggiuntive. 4. Utilizzate il NDepend rapporto del passaggio precedente per identificare tutti i file richiesti. Copia questi file dall'applicazione che hai aggiornato in precedenza e aggiungili alla nuova soluzione. 5. Crea la soluzione e risolvi tutti i problemi. Per ulteriori informazioni sulla creazione di progetti e soluzioni, consulta la documentazione di Visual Studio.	Sviluppatore di app

Operazione	Description	Competenze richieste
	Note Man mano che si crea la soluzione e si verifica la funzionalità, è possibile identificare diversi file aggiuntiv i da aggiungere alla soluzione, oltre ai file NDepend identificati.	

Aggiorna il codice dell'applicazione

Operazione	Description	Competenze richieste
Implementa APIs il web con ASP.NET Core.	Supponiamo che uno dei microservizi che hai identific ato nella tua precedente applicazione per il carrello degli acquisti Monolith sia Products. Hai creato un nuovo progetto API web ASP.NET Core per Products nell'epopea precedente. In questo passaggio, identific hi e modernizzi tutti i moduli Web (pagine.aspx) correlati ai prodotti. Supponiamo che Products sia composto da quattro moduli Web, come illustrato in precedenza nella sezione Architettura: • Elenca prodotti	Sviluppatore di app

Operazione	Description	Competenze richieste
	• Visualizza prodotto • Aggiungi/Modifica prodotto • Elimina prodotto È necessario analizzare ogni modulo Web, identificare tutte le richieste inviate al database per eseguire una logica e ottenere risposte. È possibile implementare ogni richiesta come endpoint dell'API Web. Dati i suoi moduli web, Products può avere i seguenti endpoint possibili: • `/api/products` • `/api/products/{id}` • `/api/products/add` • `/api/products/update/{id}` • `/api/products/delete/{id}` Come accennato in precedenza, puoi anche riutilizzare tutti gli altri progetti che hai aggiornato a.NET 5, inclusi Business Logic, Data Access e i progetti. shared/common	

Operazione	Description	Competenze richieste
Configura Amazon CloudWatch Logs.	Puoi usare Amazon CloudWatch Logs per monitorare, archiviare e accedere ai log della tua applicazione. Puoi accedere ai dati in Amazon CloudWatch Logs utilizzando un SDK AWS. Puoi anche integrare applicazioni.NET con CloudWatch Logs utilizzando i più diffusi framework di logging.NET come NLogLog4Net e il framework di registrazione ASP.NET Core. Per ulteriori informazioni su questo passaggio, consulta il post di blog Amazon CloudWatch Logs e.NET Logging Frameworks.	Sviluppatore di app

Operazione	Description	Competenze richieste
Configura AWS Systems Manager Parameter Store.	Puoi utilizzare AWS Systems Manager Parameter Store per archiviare le impostazioni dell'applicazione, come le stringhe di connessione, separatamente dal codice dell'applicazione. Il NuGet pacchetto Amazon.Extensions.Configuration.SystemsManagersemplifica il modo in cui l'applicazione carica queste impostazioni da AWS Systems Manager Parameter Store nel sistema di configurazione .NET Core. Per ulteriori informazioni su questo passaggio, consulta il post del blog .NET Core configuration provider for AWS Systems Manager.	Sviluppatore di app

Aggiungi autenticazione e autorizzazione

Operazione	Description	Competenze richieste
Utilizza un cookie condiviso per l'autenticazione.	La modernizzazione di un'applicazione monolitica legacy è un processo iterativo e richiede la coesistenza del monolito e della sua versione modernizzata. È possibile utilizzare un cookie condiviso per ottenere un'autenticazione	Sviluppatore di app

Operazione	Description	Competenze richieste
	senza interruzioni tra le due versioni. L'applicazione ASP.NET legacy continua a convalidare le credenzia li dell'utente ed emette il cookie, mentre l'applicazione ASP.NET Core modernizzata convalida il cookie. Per istruzioni e codice di esempio, consulta il progetto di esempio. GitHub	

Crea ed esegui il contenitore localmente

Operazione	Description	Competenze richieste
Crea un'immagine Docker usando Visual Studio.	In questo passaggio, crei un file Docker utilizzando l'API web di Visual Studio for .NET Core. 1. Apri Visual Studio. 2. In Solution Explorer, dal menu contestuale (con il tasto destro del mouse) del progetto, scegli Aggiungi, Docker Support. 3. Seleziona Linux come sistema operativo di destinazione. Visual Studio crea un file Docker per il tuo progetto. Per	Sviluppatore di app

Operazione	Description	Competenze richieste
	un file Docker di esempio, vedi Visual Studio Container Tools for Docker sul sito Web di Microsoft.	

Operazione	Description	Competenze richieste

Crea ed esegui il contenitore utilizzando Docker Desktop.	Ora puoi creare, creare ed eseguire il contenitore in Docker Desktop.
	1. Apri una finestra del prompt dei comandi. Vai alla cartella della soluzione in cui si trova il file Docker. Esegui il seguente comando per creare l'immagine Docker:
	`docker build -t aspnetcorewebapiimage -f Dockerfile .`
	2. Esegui il comando seguente per visualizzare tutte le immagini Docker:
	`docker images`
	3. Esegui il comando seguente per creare ed eseguire un contenitore:
	`docker run -d -p 8080:80 --name aspnetcorewebapicontainer aspnetcorewebapiimage`
	4. Apri Docker Desktop, quindi scegli Contenitori/App. Puoi vedere un nuovo contenitore chiamato running.
		Sviluppatore di app

Operazione	Description	Competenze richieste
	`aspnetcorewebapicontainer`	

Risorse correlate

- Esegui un contenitore Docker dell'API Web ASP.NET Core su un'istanza Amazon EC2 Linux (AWS Prescriptive Guidance)
- Workshop Amazon ECS
- Esegui blue/green implementazioni ECS utilizzando CodeDeploy AWS (documentazione CloudFormation AWS) CloudFormation
- Guida introduttiva a (documentazione NDepend) NDepend
- Porting Assistant per.NET

Informazioni aggiuntive

Le tabelle seguenti forniscono esempi di progetti di esempio per un'applicazione legacy per il carrello degli acquisti e i progetti equivalenti nell'applicazione ASP.NET Core modernizzata.

Soluzione legacy:

Project name (Nome progetto)	Modello di progetto	Framework Target
Interfaccia aziendale	Libreria di classi	.NET Framework
BusinessLogic	Libreria di classi	.NET Framework
WebApplication	Applicazione Web ASP.NET Framework	.NET Framework
UnitTests	NUnit Progetto di test	.NET Framework
Condiviso -> Comune	Libreria di classi	.NET Framework
Condiviso -> Framework	Libreria di classi	.NET Framework

Nuova soluzione:

Project name (Nome progetto)	Modello di progetto	Framework Target
BusinessLogic	Libreria di classi	.NET 5.0
<WebAPI>	API Web ASP.NET Core	.NET 5.0
<WebAPI>. UnitTests	NUnit Progetto 3 Test	.NET 5.0
Condiviso -> Comune	Libreria di classi	.NET 5.0
Condiviso -> Framework	Libreria di classi	.NET 5.0

Esegui carichi di lavoro pianificati e basati su eventi su larga scala con AWS Fargate

RAJAGOPAL HARI OHM PRASATH, Amazon Web Services

Riepilogo

Questo modello descrive come eseguire carichi di lavoro pianificati e basati su eventi su larga scala sul cloud Amazon Web Services (AWS) utilizzando AWS Fargate.

Nel caso d'uso impostato da questo modello, il codice viene scansionato alla ricerca di informazioni sensibili di AWS, come il numero di account e le credenziali AWS, ogni volta che viene inviata una pull request. La pull request avvia una funzione Lambda. La funzione Lambda richiama un'attività Fargate che si occupa della scansione del codice. Lambda viene avviata ogni volta che viene generata una nuova pull request. Se la scansione rileva informazioni sensibili, Amazon Simple Notification Service (Amazon SNS) invia i risultati della scansione in un messaggio e-mail.

Questo modello è utile nei seguenti casi d'uso aziendali:

- Se la tua azienda deve eseguire molti carichi di lavoro pianificati e basati su eventi che non possono essere eseguiti da AWS Lambda a causa delle limitazioni relative al runtime (un limite di 15 minuti) o alla memoria
- Se desideri che AWS gestisca le istanze fornite per questi carichi di lavoro

Quando si utilizza questo modello, è possibile creare un nuovo cloud privato virtuale (VPC). Questo modello utilizza AWS CodeCommit anche.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- AWS CodeCommit per l'hosting della base di codice e la creazione di richieste pull
- AWS Command Line Interface (AWS CLI) versione 1.7 o successiva, installata e configurata su macOS, Linux o Windows
- Carichi di lavoro in esecuzione in contenitori
- Eseguibile Apache Maven configurato in classpath

Architecture

Il flusso complessivo include i seguenti passaggi.

1. Ogni volta che viene inviata una nuova pull request CodeCommit, viene avviata una funzione Lambda. La funzione Lambda ascolta l'evento `CodeCommit Pull Request State Change` tramite Amazon. EventBridge
2. La funzione Lambda invia una nuova attività Fargate con i seguenti parametri di ambiente per il controllo del codice e la scansione.

   ```
   RUNNER # <<TaskARN>>
   SNS_TOPIC # <<SNSTopicARN>>
   SUBNET # <<Subnet in which Fargate task gets launched>>
   ```

   Se la scansione rileva informazioni sensibili nel codice, Fargate invia un nuovo messaggio all'argomento Amazon SNS.
3. Un abbonato SNS legge il messaggio dall'argomento e invia un messaggio e-mail.

Tecnologia

- AWS CodeCommit
- Amazon Elastic Container Registry (Amazon ECR)
- Amazon Elastic Container Service (Amazon ECS)
- Amazon EventBridge
- AWS Fargate
- AWS Lambda
- Amazon SNS
- Docker

Tools (Strumenti)

Strumenti

- AWS CLI: AWS Command Line Interface (CLI) è uno strumento unificato per gestire i servizi AWS.

- AWS CodeCommit: AWS CodeCommit è un servizio di controllo del codice sorgente completamente gestito che ospita repository sicuri basati su Git. Utilizzando CodeCommit, i team possono collaborare sul codice in un ambiente sicuro e altamente scalabile.
- Amazon ECR — Amazon Elastic Container Registry (Amazon ECR) è un registro completamente gestito che gli sviluppatori possono utilizzare per archiviare, gestire e distribuire immagini di container Docker.
- Amazon ECS — Amazon Elastic Container Service (Amazon ECS) è un servizio di gestione dei container veloce e altamente scalabile. Puoi usare Amazon ECS per eseguire, arrestare e gestire i contenitori su un cluster.
- AWS Fargate — AWS Fargate è una tecnologia che puoi usare con Amazon ECS per eseguire container senza dover gestire server o cluster di istanze Amazon. EC2
- AWS Lambda: AWS Lambda è un servizio di elaborazione che supporta l'esecuzione di codice senza effettuare il provisioning o la gestione di server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo.
- Amazon SNS — Amazon Simple Notification Service (Amazon SNS) è un servizio gestito che fornisce il recapito dei messaggi dagli editori agli abbonati (noti anche come produttori e consumatori). Gli editori comunicano in modo asincrono con gli abbonati creando e inviando messaggi a un argomento, che rappresenta un punto di accesso logico e un canale di comunicazione. I client che sottoscrivono l'argomento SNS ricevono messaggi pubblicati utilizzando un protocollo supportato, ad esempio Lambda, e-mail, notifiche push mobili e messaggi di testo mobili (SMS).
- Docker: Docker ti aiuta a creare, testare e distribuire applicazioni in pacchetti chiamati contenitori.
- Client Git: riga di comando o strumento desktop per controllare gli artefatti richiesti
- Maven — Apache Maven è uno strumento di gestione dei progetti per la gestione centralizzata della compilazione, del reporting e della documentazione di un progetto.

Epiche

Configura il repository locale

Operazione	Description	Competenze richieste
Scarica il codice.	Nella sezione Allegati, scarica il file.zip ed estrai i file.	Sviluppatore, amministratore di sistema AWS

Operazione	Description	Competenze richieste
Configura il repository.	Esegui `mvn clean install` sulla cartella principale.	Sviluppatore, amministratore di sistema AWS

Crea un'immagine Amazon ECR e invia l'immagine

Operazione	Description	Competenze richieste
Crea un repository Amazon ECR e accedi.	Apri la console di Amazon ECR. Nel pannello di navigazione, scegli Repository, quindi scegli Crea repository. Per informazioni su questa e altre storie, consulta la sezione Risorse correlate.	Sviluppatore, amministratore di sistema AWS
Invia l'immagine del conteiner.	Apri il repository, scegli Visualizza comandi push e accedi a Docker. Dopo aver effettuato l'accesso, esegui i comandi, con le sostituzioni richieste, che si trovano in Invia l'immagine del contenitore nella sezione Informazioni aggiuntive. Questo carica l'immagine del contenitore Docker che viene utilizzata per eseguire la scansione del codice. Una volta completato il caricamento, copia l'URL dell'ultima build nel repository Amazon ECR.	Sviluppatore, amministratore di sistema AWS

Crea il CodeCommit repository

Operazione	Description	Competenze richieste
Crea il CodeCommit repository.	Per creare un nuovo CodeCommit repository AWS, esegui il comando sotto Crea il CodeCommit repository nella sezione Informazioni aggiuntive.	Sviluppatore, amministratore di sistema AWS

Crea il VPC (opzionale)

Operazione	Description	Competenze richieste
Crea un VPC.	Se desideri utilizzare un nuovo VPC anziché uno esistente, esegui i comandi in Crea un VPC nella sezione Informazioni aggiuntive. Lo script AWS Cloud Development Kit (AWS CDK) genererà il IDs VPC e la sottorete che sono stati creati.	Sviluppatore, amministratore di sistema AWS

Crea il cluster Amazon ECS e il task Fargate

Operazione	Description	Competenze richieste
Crea il cluster e l'attività.	Per creare un cluster Amazon ECS e una definizione di attività Fargate, esegui i comandi in Crea cluster e attività nella sezione Informazioni aggiuntive. Assicurati che l'ID VPC e l'URI del repository Amazon ECR corretti vengano	Sviluppatore, amministratore di sistema AWS

Operazione	Description	Competenze richieste
	passati come parametro durante l'esecuzione dello script di shell. Lo script crea una definizione di attività Fargate che punta all'immag ine Docker (responsabile della scansione). Lo script crea quindi un lavoro e un ruolo di esecuzione associato.	
Verifica il cluster Amazon ECS.	Apri la console Amazon ECS. Nel riquadro di navigazio ne, scegli Clusters e scegli il cluster Amazon ECS appena creato denominato Fargate-Job-Cluster. Dopodiché, scegli Definizione dell'atti vità nel riquadro di navigazio ne e conferma che sia presente una nuova definizio ne di attività con il prefisso. `awscdkfargateecsTa skDef`	Sviluppatore, amministratore di sistema AWS

Crea l'argomento e l'abbonato SNS

Operazione	Description	Competenze richieste
Creare un argomento SNS.	Per creare un argomento SNS, esegui il comando sotto Crea l'argomento SNS nella sezione Informazioni aggiuntiv e. Una volta completata la creazione, nota il`SNS`	Sviluppatore, amministratore di sistema AWS

Operazione	Description	Competenze richieste
	ARN, che viene utilizzato nel passaggio successivo.	
Crea l'abbonato SNS.	Per creare un abbonato e-mail per l'argomento SNS, esegui il comando in Crea l'abbonato SNS nella sezione Informazioni aggiuntive. Assicurati di sostituirlo `TopicARN` e `Email address` utilizzarlo nel comando CLI. Per ricevere notifiche e-mail, assicurati di confermare l'indirizzo e-mail utilizzato come abbonato.	Sviluppatore, amministratore di sistema AWS

Creare la funzione Lambda e il trigger CodeCommit

Operazione	Description	Competenze richieste
Crea la funzione e il trigger.	Per creare una funzione Lambda con un CodeCommit trigger, esegui il comando in Funzione Lambda e CodeCommit trigger nella sezione Informazioni aggiuntive. Assicurati di sostituire i parametri con i valori corrispondenti prima di eseguire il comando. Lo script crea la funzione Lambda e la configura per essere invocata quando viene effettuata una nuova richiesta pull.	Sviluppatore, amministratore di sistema AWS

Eseguire il test dell'applicazione

Operazione	Description	Competenze richieste
Testare l'applicazione.	Se si archiviano informazioni sensibili di AWS nel CodeCommit repository, è necessario avviare la funzione Lambda. La funzione Lambda avvia l'attività Fargate, che esegue la scansione del codice e invia i risultati della scansione in una notifica e-mail.	Sviluppatore, amministratore di sistema AWS

Risorse correlate

- Creazione di un repository Amazon ECR
- Trasferimento di immagini Docker ad Amazon ECR

Informazioni aggiuntive

Invia l'immagine del contenitore

```
> cd 1-ecr-image-push
> ./run.sh <<ecr-repository>>
```

Crea il CodeCommit repository

```
aws codecommit create-repository --repository-name test-repo --repository-description
 "My Test repository"
```

Crea un VPC

```
> cd 2-create-vpc
> ./run.sh
```

Output

```
aws-batch-cdk-vpc-efs-launch-template.privatesubnet = subnet-<<id>>
aws-batch-cdk-vpc-efs-launch-template.publicsubnet = subnet-<<id>>
aws-batch-cdk-vpc-efs-launch-template.vpcid = vpc-<<id>>
```

Crea il cluster e l'attività

```
> export CDK_DEFAULT_ACCOUNT = <<aws_account_id>>
> export CDK_DEFAULT_REGION = <<aws_region>>
> cd 3-create-ecs-task
> ./run.sh <<vpc-id>> <<ecr-repo-uri>>
```

Output

```
aws-cdk-fargate-ecs.CLUSTERNAME = Fargate-Job-Cluster
aws-cdk-fargate-ecs.ClusterARN = <<cluster_arn>>
aws-cdk-fargate-ecs.ContainerARN = Fargate-Container
aws-cdk-fargate-ecs.TaskARN = <<task_arn>>
aws-cdk-fargate-ecs.TaskExecutionRole = <<execution_role_arn>>
aws-cdk-fargate-ecs.TaskRole = <<task_role_arn>>
```

Crea l'argomento SNS

```
aws sns create-topic --name code-commit-topic
```

Crea l'abbonato SNS

```
aws sns subscribe \
    --topic-arn <<topic_arn>> \
    --protocol email \
    --notification-endpoint <<email_address>>
```

Funzione Lambda e grilletto CodeCommit

```
> export CDK_DEFAULT_ACCOUNT = <<aws_account_id>>
> export CDK_DEFAULT_REGION = <<aws_region>>
> cd 5-Lambda-CodeCommit-Trigger
> ./run.sh <<taskarn>> <<snstopicarn>> subnet-<<id>> <<codecommitarn>>
```

Output

```
aws-cdk-fargate-lambda-event.Cloudwatchrule = <<cloudwatchrule>>
aws-cdk-fargate-lambda-event.CodeCommitLambda = AWS-Code-Scanner-Function
aws-cdk-fargate-lambda-event.LambdaRole = <<lambdaiamrole>>
```

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Onboarding dei tenant nell'architettura SaaS per il modello a silo utilizzando C# e AWS CDK

Tabby Ward, Susmitha Reddy Gankidi e Vijai Anand Ramalingam, Amazon Web Services

Riepilogo

Le applicazioni Software as a Service (SaaS) possono essere create con una varietà di modelli architettonici diversi. Il modello a silo si riferisce a un'architettura in cui ai tenant vengono fornite risorse dedicate.

Le applicazioni SaaS si basano su un modello semplice per introdurre nuovi inquilini nel loro ambiente. Ciò richiede spesso l'orchestrazione di una serie di componenti per fornire e configurare correttamente tutti gli elementi necessari per creare un nuovo tenant. Questo processo, nell'architettura SaaS, viene chiamato onboarding dei tenant. L'onboarding dovrebbe essere completamente automatizzato per ogni ambiente SaaS utilizzando l'infrastruttura come codice nel processo di onboarding.

Questo modello ti guida attraverso un esempio di creazione di un tenant e di provisioning di un'infrastruttura di base per il tenant su Amazon Web Services (AWS). Il modello utilizza C# e AWS Cloud Development Kit (AWS CDK).

Poiché questo modello crea un allarme di fatturazione, consigliamo di implementare lo stack negli Stati Uniti orientali (Virginia settentrionale) o us-east-1, nella regione AWS. Per ulteriori informazioni, consulta la documentazione di AWS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un principal AWS Identity and Access Management (IAM) con accesso IAM sufficiente per creare risorse AWS per questo modello. Per maggiori informazioni, consulta Ruoli IAM.
- Installa Amazon Command Line Interface (AWS CLI) e configura AWS CLI per eseguire la distribuzione di AWS CDK.
- Visual Studio 2022 scaricato e installato o Visual Studio Code scaricato e installato.
- Configurazione di AWS Toolkit for Visual Studio.

- .NET Core 3.1 o versione successiva (richiesto per le applicazioni C# AWS CDK)
- Amazon.Lambda.Tools installato.

Limitazioni

- AWS CDK utilizza AWS CloudFormation, quindi le applicazioni AWS CDK sono soggette a quote di CloudFormation servizio. Per ulteriori informazioni, consulta AWS CloudFormation quotas.
- Lo CloudFormation stack tenant viene creato con un ruolo di CloudFormation servizio `infra-cloudformation-role` con caratteri jolly sulle azioni (`sns* esqs*`) ma con risorse limitate al prefisso. `tenant-cluster` Per un caso d'uso di produzione, valuta questa impostazione e fornisci solo l'accesso richiesto a questo ruolo di servizio. La funzione `InfrastructureProvision` Lambda utilizza anche un carattere jolly (`cloudformation*`) per effettuare il provisioning dello CloudFormation stack, ma con risorse limitate al prefisso. `tenant-cluster`
- La build docker di questo codice di esempio utilizza `--platform=linux/amd64` per forzare le immagini `linux/amd64` basate. Questo per garantire che gli artefatti dell'immagine finale siano adatti a Lambda, che per impostazione predefinita utilizza l'architettura x86-64. Se devi modificare l'architettura Lambda di destinazione, assicurati di modificare sia i codici Dockerfiles che quelli AWS CDK. Per ulteriori informazioni, consulta il post sul blog Migrazione delle funzioni AWS Lambda su processori AWS Graviton2 basati su ARM.
- Il processo di eliminazione dello stack non eliminerà i log (gruppi di CloudWatch log e log) generati dallo stack. È necessario pulire manualmente i log tramite la CloudWatch console Amazon della Console di gestione AWS o tramite l'API.

Questo modello è impostato come esempio. Per l'uso in produzione, valuta le seguenti configurazioni e apporta le modifiche in base ai requisiti aziendali:

- Il bucket AWS Simple Storage Service (Amazon S3) in questo esempio non ha il controllo delle versioni abilitato per motivi di semplicità. Valuta e aggiorna la configurazione secondo necessità.
- Questo esempio configura gli endpoint dell'API REST di Amazon API Gateway senza autenticazione, autorizzazione o limitazione per motivi di semplicità. Per l'uso in produzione, consigliamo di integrare il sistema con l'infrastruttura di sicurezza aziendale. Valuta questa impostazione e aggiungi le impostazioni di sicurezza richieste, se necessario.
- Per questo esempio di infrastruttura tenant, Amazon Simple Notification Service (Amazon SNS) e Amazon Simple Queue Service (Amazon SQS) hanno solo configurazioni minime. L'AWS Key Management Service (AWS KMS) per ogni tenant consente di utilizzare i servizi Amazon e Amazon

CloudWatch SNS dell'account in base alla policy delle chiavi AWS KMS. La configurazione è solo un esempio di segnaposto. Modifica le configurazioni secondo necessità in base al tuo caso d'uso aziendale.

- L'intera configurazione, che include, a titolo esemplificativo ma non esaustivo, gli endpoint delle API e il provisioning e l'eliminazione dei tenant di backend tramite AWS CloudFormation, copre solo il caso base di happy path. Valuta e aggiorna la configurazione con la logica di riprova necessaria, la logica di gestione degli errori aggiuntiva e la logica di sicurezza in base alle tue esigenze aziendali.
- Il codice di esempio viene testato con up-to-date cdk-nag per verificare le politiche al momento della stesura di questo documento. In futuro potrebbero essere applicate nuove politiche. Queste nuove politiche potrebbero richiedere la modifica manuale dello stack in base ai consigli prima di poter implementare lo stack. Esamina il codice esistente per assicurarti che sia in linea con i requisiti aziendali.
- Il codice si basa sul CDK AWS per generare un suffisso casuale anziché affidarsi a nomi fisici assegnati statici per la maggior parte delle risorse create. Questa configurazione serve a garantire che queste risorse siano uniche e non entrino in conflitto con altri stack. Per ulteriori informazioni, consulta la documentazione di AWS CDK. Adattalo in base alle tue esigenze aziendali.
- Questo codice di esempio impacchetta gli artefatt.NET Lambda in immagini basate su Docker e viene eseguito con il runtime dell'immagine Container fornito da Lambda. Il runtime dell'immagine del contenitore presenta vantaggi per i meccanismi standard di trasferimento e archiviazione (registri dei contenitori) e per ambienti di test locali più accurati (tramite l'immagine del contenitore). Puoi cambiare il progetto in modo che utilizzi i runtime.NET forniti da Lambda per ridurre il tempo di compilazione delle immagini Docker, ma dovrai quindi configurare meccanismi di trasferimento e archiviazione e assicurarti che la configurazione locale corrisponda alla configurazione Lambda. Modifica il codice per allinearlo ai requisiti aziendali degli utenti.

Versioni del prodotto

- AWS CDK versione 2.45.0 o successiva
- Visual Studio 2022

Architecture

Stack tecnologico

- Gateway Amazon API

- AWS CloudFormation
- Amazon CloudWatch
- Amazon DynamoDB
- AWS Identity and Access Management (IAM)
- AWS KMS
- AWS Lambda
- Simple Storage Service (Amazon S3)
- Amazon SNS
- Amazon SQS

Architettura

Il diagramma seguente mostra il flusso di creazione dello stack dei tenant. Per ulteriori informazioni sugli stack tecnologici control-plane e tenant, consultate la sezione Informazioni aggiuntive.

Flusso di creazione dello stack dei tenant

1. L'utente invia una richiesta API POST con un nuovo payload del tenant (nome del tenant, descrizione del tenant) in JSON a un'API REST ospitata da Amazon API Gateway. L'API Gateway elabora la richiesta e la inoltra alla funzione backend Lambda Tenant Onboarding. In questo esempio, non esiste alcuna autorizzazione o autenticazione. In una configurazione di produzione, questa API deve essere integrata con il sistema di sicurezza dell'infrastruttura SaaS.
2. La funzione Tenant Onboarding verifica la richiesta. Quindi tenta di archiviare il record del tenant, che include il nome del tenant, l'identificatore univoco universale (UUID) generato dal tenant e la descrizione del tenant, nella tabella di onboarding dei tenant di Amazon DynamoDB.
3. Dopo che DynamoDB ha archiviato il record, un flusso DynamoDB avvia la funzione downstream Lambda Tenant Infrastructure.
4. La funzione Tenant Infrastructure Lambda agisce in base al flusso DynamoDB ricevuto. Se lo stream è destinato all'evento INSERT, la funzione utilizza la NewImage sezione dello stream (ultimo record di aggiornamento, campo Tenant Name) per richiamare CloudFormation la creazione di una nuova infrastruttura tenant utilizzando il modello archiviato nel bucket S3. Il CloudFormation modello richiede il parametro Tenant Name.
5. AWS CloudFormation crea l'infrastruttura tenant in base al CloudFormation modello e ai parametri di input.

6. Ogni configurazione dell'infrastruttura tenant presenta un CloudWatch allarme, un allarme di fatturazione e un evento di allarme.
7. L'evento di allarme diventa un messaggio a un argomento SNS, che viene crittografato dalla chiave AWS KMS del tenant.
8. L'argomento SNS inoltra il messaggio di allarme ricevuto alla coda SQS, che viene crittografata dalla chiave di crittografia AWS KMS for encryption del tenant.

Altri sistemi possono essere integrati con Amazon SQS per eseguire azioni basate sui messaggi in coda. In questo esempio, per mantenere il codice generico, i messaggi in arrivo rimangono in coda e richiedono l'eliminazione manuale.

Flusso di eliminazione dello stack dei tenant

1. L'utente invia una richiesta API DELETE con un nuovo payload del tenant (nome del tenant, descrizione del tenant) in JSON all'API REST ospitata da Amazon API Gateway, che elaborerà la richiesta e la inoltrerà alla funzione Tenant Onboarding. In questo esempio, non esiste alcuna autorizzazione o autenticazione. In una configurazione di produzione, questa API sarà integrata con il sistema di sicurezza dell'infrastruttura SaaS.
2. La funzione Tenant Onboarding verificherà la richiesta e quindi tenterà di eliminare il record del tenant (nome del tenant) dalla tabella Tenant Onboarding.
3. Dopo che DynamoDB ha eliminato correttamente il record (il record esiste nella tabella e viene eliminato), un flusso DynamoDB avvia la funzione downstream Lambda Tenant Infrastructure.
4. La funzione Tenant Infrastructure Lambda agisce in base al record di flusso DynamoDB ricevuto. Se lo stream è destinato all'evento REMOVE, la funzione utilizza la OldImage sezione del record (informazioni sul record e campo Tenant Name, prima dell'ultima modifica, ovvero delete) per avviare l'eliminazione di uno stack esistente in base alle informazioni di quel record.
5. AWS CloudFormation elimina lo stack di tenant di destinazione in base all'input.

Tools (Strumenti)

Servizi AWS

- Amazon API Gateway ti aiuta a creare, pubblicare, gestire, monitorare e proteggere REST, HTTP e WebSocket APIs su qualsiasi scala.
- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software che aiuta a definire e fornire l'infrastruttura cloud AWS in codice.

- AWS CDK Toolkit è un kit di sviluppo cloud a riga di comando che ti aiuta a interagire con l'app AWS Cloud Development Kit (AWS CDK).
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.
- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account e regioni AWS.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.
- Amazon Simple Queue Service (Amazon SQS) fornisce una coda ospitata sicura, durevole e disponibile che ti aiuta a integrare e disaccoppiare sistemi e componenti software distribuiti.
- AWS Toolkit for Visual Studio è un plug-in per l'ambiente di sviluppo integrato (IDE) di Visual Studio. Il Toolkit for Visual Studio supporta lo sviluppo, il debug e la distribuzione di applicazioni.NET che utilizzano i servizi AWS.

Altri strumenti

- Visual Studio è un IDE che include compilatori, strumenti di completamento del codice, progettisti grafici e altre funzionalità che supportano lo sviluppo del software.

Codice

Il codice per questo pattern si trova nel repository Tenant onboarding in SaaS Architecture for Silo Model APG Example.

Epiche

Configura AWS CDK

Operazione	Description	Competenze richieste
Verifica l'installazione di Node.js.	Per verificare che Node.js sia installato sul computer locale, esegui il comando seguente. `node --version`	Amministratore AWS, AWS DevOps
Installa AWS CDK Toolkit.	Per installare AWS CDK Toolkit sul tuo computer locale, esegui il comando seguente. `npm install -g aws-cdk` Se npm non è installato, puoi installarlo dal sito Node.js.	Amministratore AWS, AWS DevOps
Verifica la versione di AWS CDK Toolkit.	Per verificare che la versione di AWS CDK Toolkit sia installata correttamente sul computer, esegui il comando seguente. `cdk --version`	Amministratore AWS, AWS DevOps

Rivedi il codice per il piano di controllo dell'onboarding dei tenant

Operazione	Description	Competenze richieste
Clonare il repository.	Clona il repository e accedi alla `\tenant-onboarding`	Amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste
	`-in-saas-architect ure-for-silo-model- apg-example` cartella. In Visual Studio 2022, apri la `\src\TenantOnboard ingInfra.sln` soluzione . Apri il `TenantOnb oardingInfraStack. cs` file e rivedi il codice. Le seguenti risorse vengono create come parte di questo stack: • DynamoDB tabella • Bucket S3 (carica il CloudFormation modello nel bucket S3). • Ruolo di esecuzione Lambda • funzione Lambda • API Gateway API • Sorgente evento per la funzione Lambda	

Operazione	Description	Competenze richieste
Esamina il CloudFormation modello.	Nella `\tenant-onboarding-in-saas-architecture-for-silo-model-apg-example\template` cartella `infra.yaml` , apri e rivedi il CloudFormation modello. Questo modello verrà idratato con il nome del tenant recuperato dalla tabella DynamoDB di onboarding del tenant. Il modello fornisce l'infrastruttura specifica del tenant. In questo esempio, effettua il provisioning della chiave AWS KMS, Amazon SNS, Amazon SQS e dell'allarme. CloudWatch	Sviluppatore di app, AWS DevOps

Operazione	Description	Competenze richieste
Esamina la funzione di onboarding dei tenant.	Apri `Function.cs` e rivedi il codice per la funzione di onboarding dei tenant, creata con il modello Visual Studio AWS Lambda Project (.NET Core- C#) con il blueprint .NET 6 (Container Image). Apri e rivedi il codice. `Dockerfile` `Dockerfile` È un file di testo che contiene istruzioni per creare l'immagine del contenitore Lambda. Nota che i seguenti NuGet pacchetti vengono aggiunti come dipendenze al `TenantOnboardingFunction` progetto: • `Amazon.Lambda.APIGatewayEvents` • `AWSSDK.DynamoDBv2` • `Newtonsoft.Json`	Sviluppatore di app, AWS DevOps

Operazione	Description	Competenze richieste
Esamina la InfraProvisioning funzione Tenant.	Accedi a `\tenant-onboarding-in-saas-architecture-for-silo-model-apg-example\src\InfraProvisioningFunction` . Apri `Function.cs` ed esamina il codice per la funzione di provisioning dell'infrastruttura tenant, creata con il modello Visual Studio AWS Lambda Project (.NET Core- C#) con il blueprint .NET 6 (Container Image). Apri e rivedi il `Dockerfile` codice. Nota che i seguenti NuGet pacchetti vengono aggiunti come dipendenze al `InfraProvisioningFunction` progetto: • `Amazon.Lambda.DynamoDBEvents` • `AWSSDK.DynamoDBv2` • `AWSSDK.Cloudformation`	Sviluppatore di app, AWS DevOps

Implementa le risorse AWS

Operazione	Description	Competenze richieste
Crea la soluzione.	Per creare la soluzione , effettuate le seguenti operazioni: 1. In Visual Studio 2022, apri la `\tenant-onboarding-in-saas-architecture-for-silo-model-apg-example\src\TenantOnboardingInfra.sln` soluzione. 2. Apri il menu contestuale (fai clic con il pulsante destro del mouse) per la soluzione e scegli Crea soluzione. **Note** Assicurati di aggiornare il `Amazon.CDK.Lib NuGet` pacchetto alla versione più recente del `\tenant-onboarding-in-saas-architecture-for-silo-model-apg-example\src\TenantOnboardingInfra`	Sviluppatore di app

Operazione	Description	Competenze richieste
	progetto prima di creare la soluzione.	

Esegui il bootstrap dell'ambiente AWS CDK.	Apri il prompt dei comandi di Windows e accedi alla cartella principale dell'app AWS CDK in cui è disponibile il `cdk.json` file ()`\tenant-onboarding-in-saas-architecture-for-silo-model-apg-example` . Esegui il seguente comando per il bootstrap.
	`cdk bootstrap`
	Se hai creato un profilo AWS per le credenziali, usa il comando con il tuo profilo.
	`cdk bootstrap --profile <profile name>`
		Amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste

Elenca gli stack CDK AWS.	Per elencare tutti gli stack da creare nell'ambito di questo progetto, esegui il comando seguente.
	`cdk ls`
	`cdk ls --profile <profile name>`
	Se hai creato un profilo AWS per le credenziali, usa il comando con il tuo profilo.
	`cdk ls --profile <profile name>`
		Amministratore AWS, AWS DevOps

Verifica quali risorse AWS verranno create.	Per esaminare tutte le risorse AWS che verranno create nell'ambito di questo progetto, esegui il comando seguente.
	`cdk diff`
	Se hai creato un profilo AWS per le credenziali, usa il comando con il tuo profilo.
	`cdk diff --profile <profile name>`
		Amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste

Distribuisci tutte le risorse AWS utilizzando AWS CDK.	Per distribuire tutte le risorse AWS esegui il seguente comando.
	`cdk deploy --all --require-approval never`
	Se hai creato un profilo AWS per le credenziali, usa il comando con il tuo profilo.
	`cdk deploy --all --require-approval never --profile <profile name>`
	Una volta completata la distribuzione, copia l'URL dell'API dalla sezione degli output del prompt dei comandi, come mostrato nell'esempio seguente.
	`Outputs: TenantOnboardingInfraStack.TenantOnboardingAPIEndpoint42E526D7 = https://j2qmp8ds21i1i.execute-api.us-west-2.amazonaws.com/prod/`
		Amministratore AWS, AWS DevOps

Verifica la funzionalità

Operazione	Description	Competenze richieste

Crea un nuovo inquilino.	Per creare il nuovo tenant, invia la seguente richiesta curl.
	`curl -X POST <TenantOnboardingAPIEndpoint* from CDK Output>tenant -d '{"Name":"Tenant123", "Description":"Stack for Tenant123"}'`
	Cambia il `<TenantOnboardingAPIEndpoint* from CDK Output>` segnaposto con il valore effettivo di AWS CDK, come mostrato nell'esempio seguente.
	`curl -X POST https://j2qmp8ds21i1i.execute-api.us-west-2.amazonaws.com/prod/tenant -d '{"Name": "Tenant123", "Description":"test12"}'`
	L'esempio seguente mostra l'output.
	`{"message": "A new tenant added - 5/4/2022 7:11:30 AM"}`
		Sviluppatore di app, amministratore AWS, AWS DevOps

Verifica i dettagli del tenant appena creato in DynamoDB.	Per verificare i dettagli del tenant appena creato in	Sviluppatore di app, amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste
	DynamoDB, procedi nel seguente modo. 1. Apri la Console di gestione AWS e accedi al servizio Amazon DynamoDB. 2. Nella barra di navigazione a sinistra, scegli Esplora gli elementi e scegli la `TenantOnboarding` tabella. **Note** Il nome dell'inquilino verrà preceduto da. `tenantcluster-` Per ulteriori informazioni, consulta la sezione Informazioni aggiuntive. 3. Verifica che venga creato un nuovo elemento con i dettagli del tenant.	

Operazione	Description	Competenze richieste
Verifica la creazione dello stack per il nuovo tenant.	Verifica che il nuovo stack sia stato creato correttamente e dotato dell'infrastruttura per il tenant appena creato in base al modello. CloudFormation 1. Apri la console. CloudFormation 2. Nella barra di navigazione a sinistra, scegli Stacks e verifica che uno stack con il nome del tenant sia stato creato correttamente. 3. Scegli lo stack di tenant appena creato, quindi scegli la scheda Risorse. Prendi nota della risorsa di allarme e della risorsa Amazon SQS. 4. Apri un nuovo terminale con le credenziali AWS configurate e punta alla regione corretta. Per generare un allarme di prova, inserisci il codice seguente, sostituendolo `<alarm resource name>` con il nome della risorsa di allarme indicato nel passaggio 3. `aws cloudwatch set-alarm-state --alarm-name <alarm resource name> --state-value`	Sviluppatore di app, amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste
	ALARM --state-reason 'Test setup' L'esempio seguente mostra il codice con il nome di una risorsa di allarme. aws cloudwatch set-alarm-state --alarm-name tenantcluster-tenant123-alarm --state-value ALARM --state-reason 'Test setup' 5. Apri la console e accedi alla console Amazon SQS. Scegli il nome della risorsa Amazon SQS identificato nella fase 3. Segui le istruzioni della documentazione AWS per ricevere ed eliminare il messaggio di test dall'allarme generato nella fase 4.	

Operazione	Description	Competenze richieste

Elimina lo stack dei tenant.	Per eliminare lo stack dei tenant, invia la seguente richiesta curl.
	
	`curl -X DELETE <TenantOnboardingAPIEndpoint* from CDK Output>tenant/<Tenant Name from previous step>`
	
	Cambia il `<TenantOnboardingAPIEndpoint* from CDK Output>` segnaposto con il valore effettivo da AWS CDK e passa `<Tenant Name from previous step>` al valore effettivo della precedente fase di creazione del tenant, come mostrato nell'esempio seguente.
	
	`curl -X DELETE https://j2qmp8ds21i1i.execute-api.us-west-2.amazonaws.com/prod/tenant/Tenant123`
	
	L'esempio seguente mostra l'output.
	
	`{"message": "Tenant destroyed - 5/4/2022 7:14:48 AM"}`
		Sviluppatore di app, AWS DevOps, amministratore AWS

Operazione	Description	Competenze richieste
Verifica l'eliminazione dello stack per il tenant esistente.	Per verificare che lo stack di tenant esistente sia stato eliminato, effettuate le seguenti operazioni: 1. Apri la console e accedi alla CloudFormation console. 2. Nella barra di navigazione a sinistra, verifica che lo stack esistente con il nome del tenant non sia più nella console (se la CloudFormation console è configurata per mostrare solo gli stack attivi) o che sia in fase di eliminazione. Se lo stack non è più presente nella CloudFormation console, utilizza l'elenco a discesa per modificare l'impostazione della console da Attivo a Eliminato per visualizzare lo stack eliminato e verificare che lo stack sia stato eliminato correttamente.	Sviluppatore di app, amministratore AWS, AWS DevOps

Eliminazione

Operazione	Description	Competenze richieste
Distruggi l'ambiente.	Prima di ripulire lo stack, assicurati quanto segue:	Amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste
	• Tutti i record in DynamoDB vengono rimossi tramite la precedente operazione di eliminazione del tenant o tramite la console o l'API DynamoDB. L'elimina zione dei record di ogni tenant avvierà la pulizia della sua controparte AWS. CloudFormation • Tutti gli CloudFormation stack AWS basati su tenant vengono ripuliti (nel caso in cui la logica di pulizia dei trigger di DynamoDB fallisca) sulla console AWS. CloudFormation Al termine del test, è possibile utilizzare AWS CDK per distruggere tutti gli stack e le risorse correlate eseguendo il comando seguente. `cdk destroy --all;` Se hai creato un profilo AWS per le credenziali, usa il profilo. Conferma la richiesta di eliminazione dello stack per eliminare lo stack.	

Operazione	Description	Competenze richieste
Pulisci Amazon CloudWatch Logs.	Il processo di eliminazione dello stack non eliminerà CloudWatch i log (gruppi di log e log) generati dallo stack. Pulisci manualmente CloudWatch le risorse utilizzan do la CloudWatch console o l'API.	Sviluppatore di app, AWS DevOps, amministratore AWS

Risorse correlate

- Workshop AWS CDK.NET
- Lavorare con AWS CDK in C#
- Riferimento CDK.NET

Informazioni aggiuntive

Stack tecnologico Control-Plane

Il codice CDK scritto in .NET viene utilizzato per effettuare il provisioning dell'infrastruttura del piano di controllo, che comprende le seguenti risorse:

1. API Gateway

   Funge da punto di ingresso dell'API REST per lo stack del piano di controllo.

2. Funzione Lambda di onboarding dei tenant

   Questa funzione Lambda viene avviata da API Gateway utilizzando il metodo m.

   Una richiesta API del metodo POST comporta l'inserimento di (`tenant name`,`tenant description`) nella tabella `Tenant Onboarding` DynamoDB.

   In questo esempio di codice, il nome del tenant viene utilizzato anche come parte del nome dello stack del tenant e dei nomi delle risorse all'interno di tale stack. Questo serve a facilitare l'identificazione di queste risorse. Il nome del tenant deve essere univoco in tutta la configurazione

per evitare conflitti o errori. La configurazione dettagliata della convalida degli input è spiegata nella documentazione dei ruoli IAM e nella sezione Limitazioni.

Il processo di persistenza nella tabella DynamoDB avrà esito positivo solo se il nome del tenant non viene utilizzato in nessun altro record della tabella.

Il nome del tenant in questo caso è la chiave di partizione per questa tabella, poiché solo la chiave di partizione può essere utilizzata come espressione di condizione. `PutItem`

Se il nome del tenant non è mai stato registrato prima, il record verrà salvato correttamente nella tabella.

Tuttavia, se il nome del tenant è già utilizzato da un record esistente nella tabella, l'operazione avrà esito negativo e avvierà un'eccezione DynamoDB. `ConditionalCheckFailedException` L'eccezione verrà utilizzata per restituire un messaggio di errore (`HTTP BadRequest`) che indica che il nome del tenant esiste già.

Una richiesta API di `DELETE` metodo rimuoverà il record per un nome di tenant specifico dalla tabella `Tenant Onboardin` g.

L'eliminazione del record DynamoDB in questo esempio avrà esito positivo anche se il record non esiste.

Se il record di destinazione esiste e viene eliminato, creerà un record di flusso DynamoDB. In caso contrario, non verrà creato alcun record downstream.

3. Onboarding di DynamoDB da parte dei tenant, con Amazon DynamoDB Streams abilitato

   Questo registra le informazioni sui metadati del tenant e qualsiasi salvataggio o eliminazione di record invierà un flusso a valle alla funzione `Tenant Infrastructure` Lambda.

4. Funzione Lambda dell'infrastruttura tenant

   Questa funzione Lambda viene avviata dal record di flusso DynamoDB del passaggio precedente. Se il record riguarda un `INSERT` evento, richiama AWS per CloudFormation creare una nuova infrastruttura tenant con il CloudFormation modello archiviato in un bucket S3. Se il record è for`REMOVE`, avvia l'eliminazione di uno stack esistente in base al campo del record dello stream. `Tenant Name`

5. Bucket S3

   Serve per memorizzare il modello. CloudFormation

6. Ruoli IAM per ogni funzione Lambda e un ruolo di servizio per CloudFormation

   Ogni funzione Lambda ha il suo ruolo IAM unico con autorizzazioni con privilegi minimi per svolgere il proprio compito. Ad esempio, la funzione `Tenant On-boarding` Lambda ha read/write accesso a DynamoDB e la funzione `Tenant Infrastructure` Lambda può leggere solo il flusso DynamoDB.

   Viene creato un ruolo di CloudFormation servizio personalizzato per il provisioning dello stack dei tenant. Questo ruolo di servizio contiene autorizzazioni aggiuntive per il provisioning CloudFormation dello stack (ad esempio, la chiave AWS KMS). Questo divide i ruoli tra Lambda CloudFormation ed evita tutte le autorizzazioni su un singolo ruolo (ruolo Infrastructure Lambda).

   Le autorizzazioni che consentono azioni potenti (come la creazione e l'eliminazione di CloudFormation pile) sono bloccate e consentite solo per le risorse che iniziano con. `tenantcluster-` L'eccezione è AWS KMS, a causa della sua convenzione di denominazione delle risorse. Il nome del tenant importato dall'API verrà aggiunto `tenantcluster-` insieme ad altri controlli di convalida (alfanumerico con solo trattino e limitato a meno di 30 caratteri per adattarsi alla maggior parte dei nomi delle risorse AWS). Ciò garantisce che il nome del tenant non comporti accidentalmente l'interruzione degli stack o delle risorse dell'infrastruttura di base.

Stack tecnologico Tenant

Un CloudFormation modello è archiviato nel bucket S3. Il modello fornisce la chiave AWS KMS specifica del tenant, un CloudWatch allarme, un argomento SNS, una coda SQS e una policy SQS.

La chiave AWS KMS viene utilizzata per la crittografia dei dati da Amazon SNS e Amazon SQS per i loro messaggi. Le pratiche di sicurezza per AwsSolutionsSNS2 e AwsSolutions SQS2 consigliano di configurare Amazon SNS e Amazon SQS con crittografia. Tuttavia, gli CloudWatch allarmi non funzionano con Amazon SNS quando si utilizza una chiave gestita da AWS, quindi in questo caso è necessario utilizzare una chiave gestita dal cliente. Per ulteriori informazioni, consulta l'AWS Knowledge Center.

La policy SQS viene utilizzata nella coda Amazon SQS per consentire all'argomento SNS creato di recapitare il messaggio alla coda. Senza la policy SQS, l'accesso verrà negato. Per ulteriori informazioni, consulta la documentazione di Amazon SNS.

Scomponi i monoliti in microservizi utilizzando CQRS e l'event sourcing

Rodolfo Junior Cerrada, Dmitry Gulin e Tabby Ward, Amazon Web Services

Riepilogo

Questo modello combina due modelli, utilizzando sia il pattern CQRS (Command Query Responsibility Separation) che il pattern di approvvigionamento degli eventi. Il pattern CQRS separa le responsabilità dei modelli di comando e di interrogazione. Il pattern di approvvigionamento degli eventi sfrutta la comunicazione asincrona basata sugli eventi per migliorare l'esperienza utente complessiva.

Puoi utilizzare i servizi CQRS e Amazon Web Services (AWS) per mantenere e scalare ogni modello di dati in modo indipendente, rifattorizzando al contempo la tua applicazione monolitica in un'architettura di microservizi. È quindi possibile utilizzare il pattern di approvvigionamento degli eventi per sincronizzare i dati dal database dei comandi al database delle query.

Questo modello utilizza un codice di esempio che include un file di soluzione (*.sln) che è possibile aprire utilizzando l'ultima versione di Visual Studio. L'esempio contiene il codice API Reward per mostrare come funzionano CQRS e l'event sourcing nelle applicazioni serverless e tradizionali o locali di AWS.

Per ulteriori informazioni su CQRS e l'approvvigionamento di eventi, consulta la sezione Informazioni aggiuntive.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Amazon CloudWatch
- Tabelle Amazon DynamoDB
- Amazon DynamoDB Streams
- Chiave di accesso e chiave segreta di AWS Identity and Access Management (IAM); per ulteriori informazioni, guarda il video nella sezione Risorse correlate
- AWS Lambda
- Familiarità con Visual Studio

- Familiarità con AWS Toolkit for Visual Studio; per ulteriori informazioni, guarda il video dimostrativo di AWS Toolkit for Visual Studio nella sezione Risorse correlate

Versioni del prodotto

- Visual Studio 2019 Community Edition.
- AWS Toolkit per Visual Studio 2019.
- .NET Core 3.1. Questo componente è un'opzione nell'installazione di Visual Studio. Per includere.NET Core durante l'installazione, seleziona lo sviluppo multipiattaforma NET Core.

Limitazioni

- Il codice di esempio per un'applicazione locale tradizionale (API Web ASP.NET Core e oggetti di accesso ai dati) non viene fornito con un database. Tuttavia, viene fornito con l'oggetto `CustomerData` in memoria, che funge da database fittizio. Il codice fornito è sufficiente per testare il modello.

Architecture

Stack di tecnologia di origine

- Progetto API Web ASP.NET Core
- Server Web IIS
- Oggetto di accesso ai dati
- Modello CRUD

Architettura di origine

Nell'architettura di origine, il modello CRUD contiene interfacce di comando e di interrogazione in un'unica applicazione. Per un esempio di codice, vedere `CustomerDAO.cs` (allegato).

Stack tecnologico Target

- Amazon DynamoDB
- Amazon DynamoDB Streams

- AWS Lambda
- (Opzionale) Amazon API Gateway
- (Opzionale) Amazon Simple Notification Service (Amazon SNS)

Architettura Target

Nell'architettura di destinazione, le interfacce di comando e di interrogazione sono separate. L'architettura mostrata nel diagramma seguente può essere estesa con API Gateway e Amazon SNS. Per ulteriori informazioni, consulta la sezione Informazioni aggiuntive.

1. Le funzioni Command Lambda eseguono operazioni di scrittura, come creare, aggiornare o eliminare, sul database.
2. Le funzioni Query Lambda eseguono operazioni di lettura, come get o select, sul database.
3. Questa funzione Lambda elabora i flussi DynamoDB dal database Command e aggiorna il database Query per le modifiche.

Tools (Strumenti)

Strumenti

- Amazon DynamoDB — Amazon DynamoDB è un servizio di database NoSQL completamente gestito che offre prestazioni veloci e prevedibili con una scalabilità perfetta.
- Amazon DynamoDB Streams — DynamoDB Streams acquisisce una sequenza ordinata nel tempo di modifiche a livello di elemento in qualsiasi tabella DynamoDB. Quindi memorizza queste informazioni in un registro per un massimo di 24 ore. La crittografia a riposo crittografa i dati in DynamoDB Streams.
- AWS Lambda: AWS Lambda è un servizio di elaborazione che supporta l'esecuzione di codice senza effettuare il provisioning o la gestione di server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo. Verrà addebitato soltanto il tempo di calcolo consumato e non verrà addebitato alcun costo quando il codice non è in esecuzione.
- Console di gestione AWS: la Console di gestione AWS è un'applicazione Web che comprende un'ampia raccolta di console di servizio per la gestione dei servizi AWS.
- Visual Studio 2019 Community Edition — Visual Studio 2019 è un ambiente di sviluppo integrato (IDE). La Community Edition è gratuita per i contributori open source. In questo modello, utilizzerai

Visual Studio 2019 Community Edition per aprire, compilare ed eseguire codice di esempio. Solo per la visualizzazione, puoi utilizzare qualsiasi editor di testo o Visual Studio Code.

- AWS Toolkit for Visual Studio — AWS Toolkit for Visual Studio è un plug-in per l'IDE di Visual Studio. AWS Toolkit for Visual Studio semplifica lo sviluppo, il debug e la distribuzione di applicazioni.NET che utilizzano i servizi AWS.

Codice

Il codice di esempio è allegato. Per istruzioni sulla distribuzione del codice di esempio, consulta la sezione Epics.

Epiche

Apri e crea la soluzione

Operazione	Description	Competenze richieste
Aprire la soluzione.	1. Scaricate il codice sorgente di esempio (`CQRS-ES Code.zip`) dalla sezione Allegati ed estraete i file. 2. Nell'IDE di Visual Studio, scegli File, Apri, Project Solution e vai alla cartella in cui hai estratto il codice sorgente. 3. Scegli AWS.apg.cqrses.sln , quindi scegli Apri. L'intera soluzione viene caricata in Visual Studio.	Sviluppatore di app
Crea la soluzione.	Apri il menu contestuale (fai clic con il pulsante destro del mouse) per la soluzione, quindi scegli Crea soluzione . Questo creerà e compilerà tutti i progetti della soluzione	Sviluppatore di app

Operazione	Description	Competenze richieste
	. Dovrebbe essere compilato correttamente. Visual Studio Solution Explorer dovrebbe mostrare la struttura delle cartelle. • `CQRS On-Premises Code Sample`contiene un esempio di utilizzo di CQRS in locale. • `CQRS AWS Serverless` contiene tutto il codice di esempio CQRS e di event-sourcing che utilizza i servizi serverless AWS.	

Crea le tabelle DynamoDB

Operazione	Description	Competenze richieste
Fornire le credenziali.	Se non disponi ancora di una chiave di accesso, guarda il video nella sezione Risorse correlate. 1. In Solution Explorer, espandi CQRS AWS Serverless, quindi espandi la cartella Build solution. 2. Espandi il progetto aws.apg.cqrses.Build e visualizza il file. `Program.cs`	Sviluppatore di app, ingegnere dei dati, DBA

Operazione	Description	Competenze richieste
	3. Scorri fino all'inizio e cerca. `Program.cs Program()` 4. `YOUR ACCESS KEY`Sostituiscilo con la chiave di accesso del tuo account e `YOUR SECRET KEY` sostituiscilo con la chiave segreta del tuo account. Tieni presente che in un ambiente di produzione non è necessari o codificare le chiavi. Invece, puoi usare AWS Secrets Manager per archiviare e recuperare le credenziali.	
Compilare il progetto.	Per creare il progetto, apri il menu contestuale (fai clic con il pulsante destro del mouse) per il progetto aws.apg.c qrses.BUILD, quindi scegli Build.	Sviluppatore di app, ingegnere dei dati, DBA
Crea e popola le tabelle.	Per creare le tabelle e popolarle con i dati iniziali, apri il menu contestuale (fai clic con il pulsante destro del mouse) per il progetto aws.apg.cqrses.Build, quindi scegli Debug, Avvia nuova istanza.	Sviluppatore di app, ingegnere dei dati, DBA

Operazione	Description	Competenze richieste
Verifica la struttura della tabella e i dati.	Per verificare, accedi ad AWS Explorer ed espandi Amazon DynamoDB. Dovrebbe visualizzare le tabelle. Apri ogni tabella per visualizzare i dati di esempio.	Sviluppatore di app, ingegnere dei dati, DBA

Esegui test locali

Operazione	Description	Competenze richieste
Costruisci il progetto CQRS.	1. Apri la soluzione e accedi alla cartella della Services/CQRS/Tests soluzione CQRS AWS. 2. In AWS.APG.CQRSES.CQRSLambda.Tests project, apri BaseFunctionTest.cs e sostituisci AccessKeye SecretKeycon le chiavi IAM che hai creato. 3. Salvare le modifiche. 4. Per compilare e creare il progetto di test, apri il menu contestuale (fai clic con il pulsante destro del mouse) del progetto, quindi scegli Build.	Sviluppatore di app, tecnico di test
Crea il progetto di event-sourcing.	1. Accedi alla cartella della soluzione CQRS AWS Services/Event Source/Tests.	Sviluppatore di app, tecnico di test

Operazione	Description	Competenze richieste
	2. Nella cartella AWS.APG.CQRSES. EventSourceLambda.Tests project, apri BaseFunctionTest.cs e sostituisci AccessKeye SecretKeycon le chiavi IAM che hai creato. 3. Salvare le modifiche. 4. Per compilare e creare il progetto di test, apri il menu contestuale (fai clic con il pulsante destro del mouse) del progetto, quindi scegli Build.	
Esegui i test.	Per eseguire tutti i test, scegli Visualizza, Test Explorer, quindi scegli Esegui tutti i test in visualizzazione. Tutti i test devono essere superati, come indicato da un'icona verde con un segno di spunta.	Sviluppatore di app, tecnico di test

Pubblica le funzioni CQRS Lambda in AWS

Operazione	Description	Competenze richieste
Pubblica la prima funzione Lambda.	1. In Solution Explorer, apri il menu contestuale (fai clic con il pulsante destro del mouse) per AWS.APG.CQRSES. CommandCreateLambda progetto,	Sviluppatore di app, DevOps ingegnere

Operazione	Description	Competenze richieste
	quindi scegli Pubblica su AWS Lambda. 2. Seleziona il profilo che desideri utilizzare e la regione AWS in cui desideri distribuire la funzione Lambda e il nome della funzione. 3. Per i campi rimanenti, mantieni i valori predefiniti e scegli Avanti. 4. Nell'elenco a discesa Nome ruolo, seleziona AWSLambdaFullAccess. 5. Per fornire le chiavi dell'account, scegli Aggiungi e inserisci `AcessKey` come variabile e la chiave di accesso come valore. Quindi scegli nuovamente Aggiungi, inserisci `SecretKey` come variabile e la tua chiave segreta come valore. 6. Per i campi rimanenti, mantieni i valori predefini ti e scegli Carica. Dopo il caricamento della funzione di test Lambda, viene visualizzata automatic amente in Visual Studio. 7. Ripeti i passaggi 1-6 per i seguenti progetti:	

Operazione	Description	Competenze richieste
	• AWS.APG.CARSES.CommandDeleteLambda • AWS.APG.CARSES.CommandUpdateLambda • AWS.APG.CARSES.CommandAddRewardLambda • AWS.APG.CARSES.CommandRedeemRewardLambda • AWS.APG.CARSES.QueryCustomerListLambda • AWS.APG.CARSES.QueryRewqardLambda	
Verifica il caricamento della funzione.	(Facoltativo) Puoi verificare che la funzione sia stata caricata correttamente accedendo ad AWS Explorer ed espandendo AWS Lambda. Per aprire la finestra di test, scegliete la funzione Lambda (doppio clic).	Sviluppatore di app, ingegnere DevOps

Operazione	Description	Competenze richieste
Prova la funzione Lambda.	1. Inserisci i dati della richiesta o copia un esempio di dati di richiesta da Dati di test nella sezione Informazioni aggiuntive. Assicurati di selezionare i dati relativi alla funzione che stai testando. 2. Per eseguire il test, scegli Invoke. La risposta e gli eventuali errori vengono visualizzati nella casella di testo Risposta, mentre i registri vengono visualizz ati nella casella di testo Registri o in Registri. CloudWatch 3. Per verificare i dati, in AWS Explorer, scegli la tabella DynamoDB (doppio clic). Tutti i progetti CQRS Lambda si trovano nelle cartelle `CQRS AWS Serverles s\CQRS\Command Microservice` e `CQRS AWS Serverless\CQRS \Command Microserv ice` solution. Per la directory e i progetti della soluzione, vedete Directory del codice sorgente nella sezione Informazioni aggiuntive.	Sviluppatore di app, DevOps ingegnere

Operazione	Description	Competenze richieste
Pubblica le funzioni rimanenti.	Ripetete i passaggi precedenti per i seguenti progetti: • AWS.APG.CARSES. CommandDeleteLambda • AWS.APG.CARSES. CommandUpdateLambda • AWS.APG.CARSES. CommandAddRewardLa mbda • AWS.APG.CARSES. CommandRedeemRewar dLambda • AWS.APG.CARSES. QueryCustomerListLambda • AWS.APG.CARSES. QueryRewqardLambda	Sviluppatore di app, DevOps ingegnere

Configura la funzione Lambda come ascoltatore di eventi

Operazione	Description	Competenze richieste
Pubblica i gestori di eventi Customer and Reward Lambda.	Per pubblicare ogni gestore di eventi, segui i passaggi dell'epopea precedente. I progetti si trovano nelle cartelle `CQRS AWS Serverless\Event Source\Customer Event` e `CQRS AWS Serverless \Event Source\Reward Event` solution. Per ulteriori informazioni, consulta la	Sviluppatore di app

Operazione	Description	Competenze richieste
	directory del codice sorgente nella sezione Informazioni aggiuntive.	

Operazione	Description	Competenze richieste
Collega il listener di eventi Lambda event-sourcing.	1. Accedi alla Console di gestione AWS utilizzando lo stesso account che usi quando pubblichi i progetti Lambda. 2. Per la regione, seleziona US East 1 o la regione in cui hai distribuito le funzioni Lambda nell'epop ea precedente. 3. Vai al servizio Lambda. 4. Seleziona la funzione `EventSourceCustome r` Lambda. 5. Scegli Aggiungi trigger. 6. Nell'elenco a discesa di configurazione Trigger, seleziona DynamoDB. 7. Nell'elenco a discesa della tabella DynamoDB, selezionare. cqrses-cu stomer-cmd 8. Nell'elenco a discesa Posizione iniziale, seleziona Taglia orizzonte da. Trim horizon significa che il trigger DynamoDB inizierà a leggere dall'ultimo record di stream (non tagliato), che è il record più vecchio dello shard. 9. Seleziona la casella di controllo Abilita trigger.	Sviluppatore di app

Operazione	Description	Competenze richieste
	10. Per i campi rimanenti, mantieni i valori predefiniti e scegli Aggiungi. Dopo che il listener è stato collegato correttamente alla tabella DynamoDB, verrà visualizzato nella pagina di progettazione Lambda.	
Pubblica e allega la funzione EventSourceReward Lambda.	Per pubblicare e allegare la funzione `EventSourceReward` Lambda, ripeti i passaggi delle due storie precedenti, selezionando cqrses-reward-cmddall'elenco a discesa della tabella DynamoDB.	Sviluppatore di app

Testa e convalida i flussi DynamoDB e il trigger Lambda

Operazione	Description	Competenze richieste
Prova lo stream e il trigger Lambda.	1. In Visual Studio, accedi ad AWS Explorer. 2. Espandi AWS Lambda e scegli la CommandReedeemRewardfunzione (doppio clic). Nella finestra della funzione che si apre, puoi testare la funzione. 3. Nella casella di testo Request, inserisci i dati della richiesta in formato	Sviluppatore di app

Operazione	Description	Competenze richieste
	JavaScript Object Notation (JSON). Per una richiesta di esempio, consulta Dati di test nella sezione Informazioni aggiuntive. 4. Scegli Richiama .	
Convalida utilizzando la tabella delle query di ricompensa di DynamodDB.	1. Apri la tabella. cqrses-reward-query 2. Controlla i punti del cliente che ha riscattato il premio. I punti riscattati devono essere sottratti dal totale dei punti aggregati del cliente.	Sviluppatore di app
Convalida, utilizzando i registri. CloudWatch	1. Vai a CloudWatch e scegli Gruppi di log. 2. Il gruppo/aws/lambda/EventSourceRewardlog contiene i log del `EventSourceReward` trigger. Tutte le chiamate Lambda vengono registrate, inclusi i messaggi inseriti `context.Logger.LogLine` e `Console.WriteLine` inseriti nel codice Lambda.	Sviluppatore di app

Operazione	Description	Competenze richieste
Convalida il trigger. EventSourceCustomer	Per convalidare il `EventSourceCustomer` trigger, ripeti i passaggi di questa epopea, utilizzando la tabella dei clienti e i registri relativi al `EventSourceCustomer` trigger. CloudWatch	Sviluppatore di app

Risorse correlate

Riferimenti

- Download di Visual Studio 2019 Community Edition
- Scarica AWS Toolkit per Visual Studio
- Guida per l'utente di AWS Toolkit for Visual Studio
- Serverless su AWS
- Casi d'uso e modelli di progettazione di DynamoDB
- Martin Fowler CQRS
- Fornitura di eventi Martin Fowler

Video

- Demo di AWS Toolkit per Visual Studio
- Come posso creare un ID di chiave di accesso per un nuovo utente IAM?

Informazioni aggiuntive

CQRS e sourcing di eventi

CQRS

Il pattern CQRS separa un singolo modello di operazioni concettuali, ad esempio un modello CRUD (create, read, update, delete) a oggetti di accesso ai dati, in modelli di operazioni di comando

e interrogazione. Il modello di comando si riferisce a qualsiasi operazione, come la creazione, l'aggiornamento o l'eliminazione, che modifica lo stato. Il modello di interrogazione si riferisce a qualsiasi operazione che restituisce un valore.

1. Il modello Customer CRUD include le seguenti interfacce:
   - `Create Customer()`
   - `UpdateCustomer()`
   - `DeleteCustomer()`
   - `AddPoints()`
   - `RedeemPoints()`
   - `GetVIPCustomers()`
   - `GetCustomerList()`
   - `GetCustomerPoints()`

Man mano che le tue esigenze diventano più complesse, puoi passare da questo approccio a modello singolo. CQRS utilizza un modello di comandi e un modello di interrogazione per separare la responsabilità della scrittura e della lettura dei dati. In questo modo, i dati possono essere mantenuti e gestiti in modo indipendente. Con una chiara separazione delle responsabilità, i miglioramenti apportati a ciascun modello non influiscono sull'altro. Questa separazione migliora la manutenzione e le prestazioni e riduce la complessità dell'applicazione man mano che cresce.

1. Interfacce nel modello Customer Command:
   - `Create Customer()`
   - `UpdateCustomer()`
   - `DeleteCustomer()`
   - `AddPoints()`
   - `RedeemPoints()`
2. Interfacce nel modello Customer Query:
   - `GetVIPCustomers()`
   - `GetCustomerList()`
   - `GetCustomerPoints()`

- `GetMonthlyStatement()`

Per un esempio di codice, vedi Directory del codice sorgente.

Il pattern CQRS quindi disaccoppia il database. Questo disaccoppiamento porta alla totale indipendenza di ogni servizio, che è l'ingrediente principale dell'architettura dei microservizi.

Utilizzando CQRS nel cloud AWS, puoi ottimizzare ulteriormente ogni servizio. Ad esempio, puoi impostare diverse impostazioni di elaborazione o scegliere tra un microservizio serverless o basato su container. Puoi sostituire la memorizzazione nella cache locale con Amazon. ElastiCache Se disponi di una publish/subscribe messaggistica locale, puoi sostituirla con Amazon Simple Notification Service (Amazon SNS). Inoltre, puoi sfruttare pay-as-you-go i prezzi e l'ampia gamma di servizi AWS che paghi solo per ciò che usi.

CQRS include i seguenti vantaggi:

- Scalabilità indipendente: ogni modello può avere la propria strategia di scalabilità adattata per soddisfare i requisiti e la domanda del servizio. Analogamente alle applicazioni ad alte prestazioni, la separazione di lettura e scrittura consente al modello di scalare in modo indipendente per soddisfare ogni esigenza. È inoltre possibile aggiungere o ridurre le risorse di elaborazione per soddisfare la richiesta di scalabilità di un modello senza influire sull'altro.
- Manutenzione indipendente: la separazione dei modelli di query e comando migliora la manutenibilità dei modelli. È possibile apportare modifiche e miglioramenti al codice di un modello senza influire sull'altro.
- Sicurezza: è più semplice applicare le autorizzazioni e le politiche a modelli separati per la lettura e la scrittura.
- Letture ottimizzate: è possibile definire uno schema ottimizzato per le query. Ad esempio, è possibile definire uno schema per i dati aggregati e uno schema separato per le tabelle dei fatti.
- Integrazione: CQRS si adatta bene ai modelli di programmazione basati su eventi.
- Complessità gestita: la separazione in modelli di query e comandi è adatta a domini complessi.

Quando utilizzate CQRS, tenete presente le seguenti avvertenze:

- Il pattern CQRS si applica solo a una parte specifica di un'applicazione e non all'intera applicazione. Se implementato in un dominio che non corrisponde al modello, può ridurre la produttività, aumentare i rischi e introdurre complessità.

- Il pattern funziona meglio per i modelli utilizzati di frequente che presentano uno squilibrio nelle operazioni di lettura e scrittura.
- Per le applicazioni che richiedono molta lettura, come i report di grandi dimensioni che richiedono tempo per l'elaborazione, CQRS offre la possibilità di selezionare il database giusto e creare uno schema per archiviare i dati aggregati. Ciò migliora il tempo di risposta di lettura e visualizzazione del report elaborando i dati del report una sola volta e scaricandoli nella tabella aggregata.
- Per le applicazioni che richiedono molta scrittura, è possibile configurare il database per le operazioni di scrittura e consentire al comando microservice di scalare in modo indipendente quando la richiesta di scrittura aumenta. Per esempi, consulta and microservices. `AWS.APG.CQRSES.CommandRedeemRewardLambda` `AWS.APG.CQRSES.CommandAddRewardLambda`

Approvvigionamento di eventi

Il passaggio successivo consiste nell'utilizzare l'origine degli eventi per sincronizzare il database delle query quando viene eseguito un comando. Ad esempio, considerate i seguenti eventi:

- Viene aggiunto un punto premio cliente che richiede l'aggiornamento dei punti premio totali o aggregati del cliente nel database delle query.
- Il cognome di un cliente viene aggiornato nel database dei comandi, il che richiede l'aggiornamento delle informazioni sostitutive sul cliente contenute nel database delle query.

Nel modello CRUD tradizionale, si garantisce la coerenza dei dati bloccandoli fino al termine di una transazione. Nell'event sourcing, i dati vengono sincronizzati mediante la pubblicazione di una serie di eventi che verranno utilizzati da un abbonato per aggiornare i rispettivi dati.

Il modello di sourcing degli eventi garantisce e registra una serie completa di azioni intraprese sui dati e le pubblica attraverso una sequenza di eventi. Questi eventi rappresentano un insieme di modifiche ai dati che i sottoscrittori di quell'evento devono elaborare per mantenere aggiornato il proprio record. Questi eventi vengono utilizzati dal sottoscrittore, sincronizzando i dati nel database del sottoscrittore. In questo caso, si tratta del database delle interrogazioni.

Il diagramma seguente mostra l'event sourcing utilizzato con CQRS su AWS.

1. Le funzioni Command Lambda eseguono operazioni di scrittura, come creare, aggiornare o eliminare, sul database.

2. Le funzioni Query Lambda eseguono operazioni di lettura, come get o select, sul database.
3. Questa funzione Lambda elabora i flussi DynamoDB dal database Command e aggiorna il database Query per le modifiche. Puoi utilizzare questa funzione anche per pubblicare un messaggio su Amazon SNS in modo che i suoi abbonati possano elaborare i dati.
4. (Facoltativo) L'abbonato all'evento Lambda elabora il messaggio pubblicato da Amazon SNS e aggiorna il database Query.
5. (Facoltativo) Amazon SNS invia una notifica e-mail dell'operazione di scrittura.

Su AWS, il database delle query può essere sincronizzato tramite DynamoDB Streams. DynamoDB acquisisce una sequenza ordinata nel tempo di modifiche a livello di elemento in una tabella DynamoDB quasi in tempo reale e archivia le informazioni in modo duraturo entro 24 ore.

L'attivazione di DynamoDB Streams consente al database di pubblicare una sequenza di eventi che rende possibile lo schema di sourcing degli eventi. Il pattern di origine degli eventi aggiunge il sottoscrittore dell'evento. L'applicazione Event Subscriber utilizza l'evento e lo elabora in base alla responsabilità del sottoscrittore. Nel diagramma precedente, il sottoscrittore dell'evento invia le modifiche al database Query DynamoDB per mantenere i dati sincronizzati. L'uso di Amazon SNS, del broker di messaggi e dell'applicazione Event Subscriber mantiene l'architettura disaccoppiata.

L'event sourcing include i seguenti vantaggi:

- Coerenza per i dati transazionali
- Una pista di controllo e una cronologia delle azioni affidabili, che possono essere utilizzate per monitorare le azioni intraprese nei dati
- Consente alle applicazioni distribuite come i microservizi di sincronizzare i dati in tutto l'ambiente
- Pubblicazione affidabile degli eventi ogni volta che lo stato cambia
- Ricostruzione o riproduzione degli stati passati
- Entità liberamente accoppiate che scambiano eventi per la migrazione da un'applicazione monolitica ai microservizi
- Riduzione dei conflitti causati dagli aggiornamenti simultanei; l'event sourcing evita la necessità di aggiornare gli oggetti direttamente nell'archivio dati
- Flessibilità ed estensibilità grazie al disaccoppiamento tra attività ed evento
- Aggiornamenti di sistema esterni
- Gestione di più attività in un unico evento

Quando utilizzi il sourcing degli eventi, tieni presente le seguenti avvertenze:

- Poiché si verifica un certo ritardo nell'aggiornamento dei dati tra i database degli abbonati di origine, l'unico modo per annullare una modifica consiste nell'aggiungere un evento di compensazione all'archivio eventi.
- L'implementazione dell'event sourcing presenta una curva di apprendimento a causa del suo diverso stile di programmazione.

Dati di test

Utilizza i seguenti dati di test per testare la funzione Lambda dopo una corretta implementazione.

CommandCreate Cliente

```
{  "Id":1501,  "Firstname":"John",  "Lastname":"Done",  "CompanyName":"AnyCompany",
 "Address": "USA",  "VIP":true }
```

CommandUpdate Cliente

```
{  "Id":1501,  "Firstname":"John",  "Lastname":"Doe",  "CompanyName":"Example Corp.",
 "Address": "Seattle, USA",  "VIP":true }
```

CommandDelete Cliente

Immettete l'ID cliente come dati della richiesta. Ad esempio, se l'ID cliente è 151, inserisci 151 come dati della richiesta.

```
151
```

QueryCustomerList

Questo campo è vuoto. Quando viene richiamato, restituirà tutti i clienti.

CommandAddReward

Ciò aggiungerà 40 punti al cliente con ID 1 (Richard).

```
{
  "Id":10101,
  "CustomerId":1,
```

```
  "Points":40
}
```

CommandRedeemReward

In questo modo verranno detratti 15 punti dal cliente con ID 1 (Richard).

```
{
  "Id":10110,
  "CustomerId":1,
  "Points":15
}
```

QueryReward

Inserisci l'ID del cliente. Ad esempio, inserisci 1 per Richard, 2 per Arnav e 3 per Shirley.

```
2
```

Directory del codice sorgente

Usa la tabella seguente come guida alla struttura di directory della soluzione Visual Studio.

Directory della soluzione CQRS On-Premises Code Sample

Modello CRUD del cliente

Esempio di codice locale CQRS\ Modello CRUD\ Progetto AWS.APG.CQRSES.DAL

Versione CQRS del modello Customer CRUD

- Comando cliente: progetto `CQRS On-Premises Code Sample\CQRS Model\Command Microservice\AWS.APG.CQRSES.Command`
- Richiesta del cliente: `CQRS On-Premises Code Sample\CQRS Model\Query Microservice\AWS.APG.CQRSES.Query` progetto

Microservizi Command and Query

Il microservizio Command si trova nella cartella della soluzione: `CQRS On-Premises Code Sample\CQRS Model\Command Microservice`

- `AWS.APG.CQRSES.CommandMicroservice`Il progetto ASP.NET Core API funge da punto di ingresso in cui i consumatori interagiscono con il servizio.
- `AWS.APG.CQRSES.Command`Il progetto.NET Core è un oggetto che ospita oggetti e interfacce relativi ai comandi.

Il microservizio di interrogazione si trova nella cartella della soluzione: `CQRS On-Premises Code Sample\CQRS Model\Query Microservice`

- `AWS.APG.CQRSES.QueryMicroservice`Il progetto ASP.NET Core API funge da punto di ingresso in cui i consumatori interagiscono con il servizio.
- `AWS.APG.CQRSES.Query`Il progetto.NET Core è un oggetto che ospita oggetti e interfacce relativi alle query.

Directory di soluzioni di codice serverless CQRS AWS

Questo codice è la versione AWS del codice locale che utilizza i servizi serverless AWS.

In C# .NET Core, ogni funzione Lambda è rappresentata da un progetto.NET Core. Nel codice di esempio di questo pattern, esiste un progetto separato per ogni interfaccia nei modelli di comando e query.

CQRS che utilizza i servizi AWS

Puoi trovare la directory della soluzione principale per CQRS che utilizza i servizi serverless AWS nella `CQRS AWS Serverless\CQRS` cartella. L'esempio include due modelli: Customer e Reward.

Le funzioni di comando Lambda per Customer e Reward si trovano nelle cartelle `CQRS\Command Microservice\Customer` e`CQRS\Command Microservice\Reward`. Contengono i seguenti progetti Lambda:

- Comando cliente: `CommandCreateLambda``CommandDeleteLambda`, e `CommandUpdateLambda`
- Comando di ricompensa: `CommandAddRewardLambda` e `CommandRedeemRewardLambda`

Le funzioni di interrogazione Lambda per Customer e Reward si trovano nelle cartelle `CQRS\Query Microservice\Customer` and`CQRS\QueryMicroservice\Reward`. Contengono i progetti `QueryCustomerListLambda` e `QueryRewardLambda` Lambda.

Progetto di test CQRS

Il progetto di test si trova nella `CQRS\Tests` cartella. Questo progetto contiene uno script di test per automatizzare il test delle funzioni CQRS Lambda.

Approvvigionamento di eventi tramite i servizi AWS

I seguenti gestori di eventi Lambda vengono avviati dai flussi DynamoDB Customer e Reward per elaborare e sincronizzare i dati nelle tabelle di query.

- La funzione `EventSourceCustomer` Lambda è mappata sul flusso `cqrses-customer-cmd` DynamoDB della tabella Customer ().
- La funzione `EventSourceReward` Lambda è mappata sul flusso `cqrses-reward-cmd` DynamoDB della tabella Reward ().

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Altri modelli

- Accedi alle applicazioni container in modo privato su Amazon EKS utilizzando AWS PrivateLink e un Network Load Balancer
- Automatizza l'aggiunta o l'aggiornamento delle voci di registro di Windows utilizzando AWS Systems Manager
- Automatizza il failover e il failback tra regioni utilizzando DR Orchestrator Framework
- Crea e distribuisci automaticamente un'applicazione Java su Amazon EKS utilizzando una pipeline CI/CD
- Crea automaticamente CI/CD pipeline e cluster Amazon ECS per microservizi utilizzando AWS CDK
- Esegui il backup e l'archiviazione dei dati del mainframe su Amazon S3 utilizzando BMC AMI Cloud Data
- Create un Micro Focus Enterprise Server PAC con Amazon EC2 Auto Scaling e Systems Manager
- Crea una rete dati aziendale con Amazon DataZone e AWS CDK AWS CloudFormation
- Containerizza i carichi di lavoro mainframe che sono stati modernizzati da Blu Age
- Converti e decomprimi i dati EBCDIC in ASCII su AWS usando Python
- Convertite file di dati mainframe con layout di registrazione complessi utilizzando Micro Focus
- Crea un portale per micro-frontend utilizzando Angular e AWS Amplify Module Federation
- Distribuisci contenitori utilizzando Elastic Beanstalk
- Emula Oracle DR utilizzando un database globale Aurora compatibile con PostgreSQL
- Genera approfondimenti sui dati utilizzando Modernizzazione del mainframe AWS Amazon Q in Quick Sight
- Genera approfondimenti sui z/OS dati Db2 utilizzando Modernizzazione del mainframe AWS Amazon Q in Quick Sight
- Identifica automaticamente le immagini duplicate dei container durante la migrazione a un repository Amazon ECR
- Implementa la diagnostica e la risoluzione dei problemi di Kubernetes basate sull'intelligenza artificiale con K8SGPT e l'integrazione con Amazon Bedrock
- Implementa l'autenticazione basata su Microsoft Entra ID in un'applicazione mainframe modernizzata AWS Blu Age
- Implementa il controllo delle versioni delle API basato sul percorso utilizzando domini personalizzati in Amazon API Gateway

- Migrazione incrementale da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL utilizzando Oracle SQL Developer e AWS SCT
- Integra il controller universale Stonebranch con la modernizzazione del mainframe AWS
- Gestisci i prodotti AWS Service Catalog in più account AWS e regioni AWS
- Esegui la migrazione di un account AWS membro da a AWS Organizations AWS Control Tower
- Esegui la migrazione e la replica di file VSAM su Amazon RDS o Amazon MSK utilizzando Connect from Precisly
- Esegui la migrazione da SAP ASE ad Amazon RDS per SQL Server utilizzando AWS DMS
- Esegui la migrazione di tabelle esterne Oracle verso Amazon Aurora, compatibile con PostgreSQL
- Modernizza l'applicazione CardDemo mainframe utilizzando AWS Transform
- Modernizza e implementa le applicazioni mainframe utilizzando e Terraform AWS Transform
- Modernizza i carichi di lavoro di stampa in batch su mainframe AWS utilizzando Rocket Enterprise Server e LRS VPSX/MFI
- Modernizza i carichi di lavoro di stampa online mainframe su AWS utilizzando Micro Focus Enterprise Server e LRS VPSX/MFI
- Modernizza la gestione dell'output del mainframe AWS utilizzando Rocket Enterprise Server e LRS X PageCenter
- Sposta i file mainframe direttamente su Amazon S3 utilizzando Transfer Family
- Ottimizza le implementazioni serverless multi-account utilizzando i flussi di lavoro e Actions AWS CDK GitHub
- Ottimizza le prestazioni della tua applicazione modernizzata AWS Blu Age
- Automatizza le blue/green implementazioni dei database globali di Amazon Aurora utilizzando i principi IaC
- Replica i database mainframe su AWS utilizzando Precisly Connect
- Esegui attività Amazon ECS su Amazon WorkSpaces con Amazon ECS Anywhere
- Invia dati di telemetria da AWS Lambda a OpenSearch per analisi e visualizzazione in tempo reale
- Imposta il rilevamento della CloudFormation deriva in un'organizzazione multiregionale e con più account
- Struttura un progetto Python in architettura esagonale utilizzando AWS Lambda
- AWS Infrastruttura di test utilizzando LocalStack e Terraform Tests
- Aggiornamento dei cluster SAP Pacemaker da a ENSA1 ENSA2
- Usa Amazon Q Developer come assistente di programmazione per aumentare la tua produttività

- Convalida il codice Account Factory for Terraform (AFT) localmente

Mainframe

Argomenti

- Accesso Servizi AWS da IBM z/OS installando AWS CLI
- Esegui il backup e l'archiviazione dei dati del mainframe su Amazon S3 utilizzando BMC AMI Cloud Data
- Crea programmi COBOL Db2 usando e Modernizzazione del mainframe AWS AWS CodeBuild
- Create un Micro Focus Enterprise Server PAC con Amazon EC2 Auto Scaling e Systems Manager
- Crea un visualizzatore di file mainframe avanzato nel cloud AWS
- Containerizza i carichi di lavoro mainframe che sono stati modernizzati da Blu Age
- Converti e decomprimi i dati EBCDIC in ASCII su AWS usando Python
- Converti i file mainframe dal formato EBCDIC al formato ASCII delimitato da caratteri in Amazon S3 utilizzando AWS Lambda
- Convertite file di dati mainframe con layout di registrazione complessi utilizzando Micro Focus
- Implementa un ambiente per applicazioni Blu Age containerizzate utilizzando Terraform
- Genera approfondimenti sui z/OS dati Db2 utilizzando Modernizzazione del mainframe AWS Amazon Q in Quick Sight
- Genera approfondimenti sui dati utilizzando Modernizzazione del mainframe AWS Amazon Q in Quick Sight
- Implementa l'autenticazione basata su Microsoft Entra ID in un'applicazione mainframe modernizzata AWS Blu Age
- Integra il controller universale Stonebranch con la modernizzazione del mainframe AWS
- Esegui la migrazione e la replica di file VSAM su Amazon RDS o Amazon MSK utilizzando Connect from Precisly
- Modernizza l'applicazione CardDemo mainframe utilizzando AWS Transform
- Modernizza e implementa le applicazioni mainframe utilizzando e Terraform AWS Transform
- Modernizza la gestione dell'output del mainframe AWS utilizzando Rocket Enterprise Server e LRS X PageCenter
- Modernizza i carichi di lavoro di stampa in batch su mainframe AWS utilizzando Rocket Enterprise Server e LRS VPSX/MFI
- Modernizzazione del mainframe: attiva DevOps AWS con Rocket Software Enterprise Suite

- Modernizza i carichi di lavoro di stampa online mainframe su AWS utilizzando Micro Focus Enterprise Server e LRS VPSX/MFI
- Sposta i file mainframe direttamente su Amazon S3 utilizzando Transfer Family
- Ottimizza le prestazioni della tua applicazione modernizzata AWS Blu Age
- Proteggi e semplifica l'accesso degli utenti in un database federativo Db2 su AWS utilizzando contesti affidabili
- Trasferisci z/OS dati Db2 su larga scala su Amazon S3 in file CSV
- Altri modelli

Accesso Servizi AWS da IBM z/OS installando AWS CLI

Souma Ghosh, Paulo Vitor Pereira e Phil de Valence, Amazon Web Services

Riepilogo

The AWS Command Line Interface (AWS CLI) è uno strumento open source per la gestione di più AWS servizi utilizzando i comandi in una shell a riga di comando. Con una configurazione minima, è possibile eseguire comandi da sessioni a riga di comando come il prompt dei comandi, il terminale e la shell bash per implementare funzionalità equivalenti a quelle fornite dal browser. Console di gestione AWS

Tutte le funzioni di amministrazione, gestione e accesso dell' AWS infrastruttura come servizio (IaaS) Console di gestione AWS sono disponibili nell' AWS API e. AWS CLIÈ possibile installarle AWS CLI su un z/OS mainframe IBM per accedere, gestire e interagire Servizi AWS direttamente da z/OS. AWS CLI Consente agli utenti e alle applicazioni di eseguire varie attività, come:

- Trasferimento di file o set di dati tra lo storage di oggetti z/OS Amazon Simple Storage Service (Amazon S3) e visualizzazione del contenuto dei bucket
- Avvio e arresto di AWS risorse diverse; ad esempio, avvio di un processo in batch in un ambiente Modernizzazione del mainframe AWS
- Chiamata di una AWS Lambda funzione per implementare una logica aziendale comune
- Integrazione con servizi di intelligenza artificiale e apprendimento automatico (AI/ML) e analisi

Questo modello descrive come installare, configurare e utilizzare AWS CLI su z/OS. You can install it globally, so it's available to all z/OS utenti o a livello di utente. Il modello descrive anche come utilizzarlo AWS CLI in una sessione interattiva a riga di comando di z/OS Unix System Services (USS) o come processo batch.

Prerequisiti e limitazioni

Prerequisiti

- Comunicazione di rete da a z/OS AWS

  Per impostazione predefinita, AWS CLI invia le richieste Servizi AWS utilizzando HTTPS sulla porta TCP 443. Per utilizzare AWS CLI correttamente, è necessario essere in grado di effettuare connessioni in uscita sulla porta TCP 443. È possibile utilizzare uno qualsiasi dei seguenti comandi

z/OS USS (alcuni di questi potrebbero non essere installati nell'ambiente in uso) per testare la connettività di rete da a: z/OS AWS

```
ping amazonaws.com
dig amazonaws.com
traceroute amazonaws.com
curl -k https://docs.aws.amazon.com/cli/v1/userguide/cli-chap-welcome.html
```

- AWS credenziali

  Per comunicare con Cloud AWS i servizi di z/OS, è AWS CLI necessario configurare alcune credenziali con privilegi di accesso alla destinazione. Account AWS Per i comandi programmatici AWS, è possibile utilizzare le chiavi di accesso, che consistono in un ID della chiave di accesso e una chiave di accesso segreta. Se non si dispone di chiavi di accesso, sarà possibile crearle dalla Console di gestione AWS. È consigliabile non utilizzare le chiavi di accesso dell'utente Account AWS root per alcuna attività a meno che non sia richiesto l'utente root. Create invece un nuovo utente IAM amministratore e preparatevi a ottenere le autorizzazioni con privilegi minimi per configurare l'utente con le chiavi di accesso. Dopo aver creato l'utente, puoi creare un ID della chiave di accesso e una chiave di accesso segreta per questo utente.

  > **Warning**
  >
  > AWS Identity and Access Management Gli utenti (IAM) dispongono di credenziali a lungo termine che presentano un rischio per la sicurezza. Per ridurre questo rischio, si consiglia di fornire a questi utenti solo le autorizzazioni necessarie per eseguire l'attività e di rimuoverli quando non sono più necessari.

- IBM Python per z/OS

  AWS CLI Richiede Python 3.8 o successivo. IBM ha consentito a Python di funzionare con IBM Open Enterprise z/OS Python for z/OS. IBM Open Enterprise Python è disponibile gratuitamente tramite Shopz SMP/E oppure è possibile scaricare il file PAX dal sito Web IBM. Per istruzioni, consulta la documentazione di installazione e configurazione per IBM Open Enterprise Python for z/OS.

Limitazioni

- Le istruzioni di installazione fornite in questo modello sono applicabili solo alla AWS CLI versione 1. L'ultima versione di AWS CLI è la versione 2. Tuttavia, questo modello utilizza la versione precedente perché i metodi di installazione sono diversi per la versione 2 e gli eseguibili binari disponibili per la versione 2 non sono compatibili con il z/OS sistema.

Versioni del prodotto

- AWS CLI versione 1
- Python 3.8 o versioni successive

Architecture

stack tecnologico

- Mainframe con z/OS
- Servizi di sistema z/OS UNIX per mainframe (USS)
- Mainframe Open MVS (OMVS): z/OS interfaccia di comando dell'ambiente shell UNIX
- Disco mainframe, ad esempio un dispositivo di archiviazione ad accesso diretto (DASD)
- AWS CLI

Architettura di destinazione

Il diagramma seguente mostra un' AWS CLI implementazione su IBM z/OS. È possibile richiamarlo AWS CLI da una sessione utente interattiva, come SSH, e da sessioni telnet. È inoltre possibile richiamarlo da un processo batch utilizzando il linguaggio di controllo del lavoro (JCL) o da qualsiasi programma in grado di chiamare un comando della shell Unix. z/OS

AWS CLI Comunica con gli Servizio AWS endpoint tramite una rete. TCP/IP Questa connessione di rete può avvenire tramite Internet o tramite una AWS Direct Connect connessione privata dal data center del cliente ai Cloud AWS data center. La comunicazione viene autenticata con AWS credenziali e crittografata.

Automazione e scalabilità

Puoi esplorare le funzionalità di an Servizio AWS with the AWS CLI e sviluppare script di shell USS per gestire AWS le tue risorse da z/OS. È inoltre possibile eseguire AWS CLI comandi e script di shell

dall'ambiente z/OS batch e automatizzare i processi in batch per eseguirli secondo una pianificazione specifica mediante l'integrazione con gli scheduler del mainframe. AWS CLI i comandi o gli script possono essere codificati all'interno di parametri (PARMs) e procedure (PROCs) e possono essere scalati seguendo l'approccio standard di chiamata al PARM o al PROC da diversi processi batch con parametri diversi.

Tools (Strumenti)

- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando.

Best practice

- Per motivi di sicurezza, limita le autorizzazioni di accesso alla directory USS in cui sono memorizzati i dettagli della chiave di AWS accesso. Consenti l'accesso solo agli utenti o ai programmi che utilizzano il AWS CLI.
- Non utilizzare le chiavi di accesso utente Account AWS root per alcuna attività. Crea invece un nuovo utente IAM amministratore e configuralo con le chiavi di accesso.

> **Warning**
>
> Gli utenti IAM dispongono di credenziali a lungo termine che presentano un rischio per la sicurezza. Per ridurre questo rischio, si consiglia di fornire a questi utenti solo le autorizza zioni necessarie per eseguire l'attività e di rimuoverli quando non sono più necessari.

Epiche

Installa AWS CLI la versione 1 su z/OS USS

Operazione	Description	Competenze richieste
Installa Python 3.8 o versione successiva.	1. Accedi all'interfaccia del prompt dei comandi z/OS USS utilizzando uno di questi metodi:	Amministratore del mainframe z/OS

Operazione	Description	Competenze richieste
	• Utilizzate un comando OMVS Time Sharing Option (TSO) dal pannello Interactive System Productivity Facility (ISPF) oppure • Utilizzate SSH o telnet per connettervi all'IP della partizione logica del mainframe (LPAR). Questo modello presuppone che `cliuser` sia l'ID utente utilizzato per accedere all'ambiente USS e `/u/cliuser/` che sia la home directory dell'utente. È possibile impostare la home directory dell'utente in modo diverso nel proprio z/OS ambiente a seconda dei requisiti di installazione. 2. Segui la guida all'installazione di IBM Open Enterprise Python for z/OS per installare Python 3.8 o versione successiva se non è già installato.	

Operazione	Description	Competenze richieste

Imposta le variabili di ambiente USS.	Aggiungi variabili di ambiente al profilo. È possibile aggiungerle al `/u/cliuser/.profile` file per un singolo utente (`cliuser`) o al `/etc/profile` file per tutti gli utenti.
	
	Note
	Questo modello presuppone che Python sia stato installato nella `/u/awscli/python` directory. Se la directory di installazione è diversa, aggiorna il codice di conseguenza.
	
	`# Python configuration`
	`export BPXKAUTOCVT='ON'`
	`export CEERUNOPTS='FILETAG(AUTOCVT,AUTOTAG) POSIX(ON)'`
	`export TAGREDIR_ERR=txt`
	`export TAGREDIR_IN=txt`
	`export TAGREDIR_OUT=txt`
	
	`# AWS CLI configuration`
		Amministratore del mainframe z/OS

Operazione	Description	Competenze richieste
	`export PATH=/u/cliuser/python/bin:$PATH` `export PYTHONPATH=/u/cliuser/python:$PYTHONPATH`	
Prova l'installazione di Python.	Esegui il comando python: `python --version` L'output dovrebbe confermare che Python 3.8 o versione successiva è installato correttamente.	Amministratore del mainframe z/OS

Operazione	Description	Competenze richieste

Verifica o installa pip.	1. Il comando pip viene in genere installato automatic amente quando si installa Python dal sito Web IBM. Per verificare, esegui il comando:
	`pip —-version`
	Se pip è installato, questo comando dovrebbe mostrare la versione installata.
	2. Se il comando pip non viene trovato, installa pip eseguendo il comando seguente:
	`python -m ensurepip --upgrade`
	Per ulteriori opzioni di installazione, consulta la documentazione pip.
		Amministratore del mainframe z/OS

Operazione	Description	Competenze richieste

Installa AWS CLI la versione 1.	1. Per installare AWS CLI, esegui il comando:
	`python -m pip install awscli`
	L'output visualizzato dovrebbe essere simile al seguente.
	`Successfully installed PyYAML-6.0.1 awscli-1.32.23 botocore-1.34.23 colorama-0.4.4 docutils-0.16 jmespath-1.0.1 pyasn1-0.5.1 python-dateutil-2.8.2 rsa-4.7.2 s3transfer-0.10.0 urllib3-2.0.7`
	2. Modifica l'autorizzazione dell'eseguibile aws eseguendo il comando seguente. Assicurati di aggiornare la directory segnaposto `<python_installation_dir>` con il percorso di installazione di Python.
	`chmod 744 <python_installation_dir>/bin/aws`
		z/OS Amministratore del mainframe

Operazione	Description	Competenze richieste

	3. Esegui il comando seguente per testare l'install azione: AWS CLI
	`aws --version`
	L'output dovrebbe mostrare le versioni di AWS CLI, Python e botocore, simili alle seguenti.
	`aws-cli/1.32.3`
	`Python/3.9.5`
	`OS/390/27.00`
	`botocore/1.34.3`	

Configura AWS CLI l'accesso da z/OS

Operazione	Description	Competenze richieste
Configura le chiavi di AWS accesso, la regione predefinita e l'output.	La AWS CLI documentazione descrive diverse opzioni per configurare AWS l'accesso. Puoi scegliere una configura zione in base agli standard della tua organizzazione. Questo esempio utilizza la configurazione delle credenzia li a breve termine. 1. Configura il AWS CLI con il seguente comando: `aws configure`	Amministratore AWS, amministratore mainframe z/OS , sviluppatore z/OS mainframe

Operazione	Description	Competenze richieste

	2. Fornisci i dettagli per i seguenti elementi quando richiesto. I valori dell'ID della chiave di accesso e della chiave di accesso segreta provengono dalle chiavi ottenute quando si impostano AWS le credenziali nei passaggi dei prerequisiti.
	
	`AWS Access Key ID [None]: ASIAIOSFODNN7EXAMPLE`
	`AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`
	`Default region name [None]: us-east-1`
	`Default output format [None]:`
	`aws configure set aws_session_token IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZVERYLONGSTRINGEXAMPLE`
	
	Questa configurazione, incluse le chiavi di accesso, è memorizzata nella `/u/cliuser/.aws` cartella. Per motivi di sicurezza, limita questa cartella per consentire l'accesso solo	

Operazione	Description	Competenze richieste
	agli utenti o ai programmi che utilizzano la AWS CLI.	
Prova il. AWS CLI	1. Esegui il comando seguente al prompt dei comandi per testarlo AWS CLI con un semplice comando: `aws s3 ls` L'output dovrebbe elencare tutti i bucket S3 configurati Account AWS senza errori. 2. Segui le istruzioni nelle prossime due epopee per trasferire dati da USS ad Amazon S3. Puoi scegliere una di queste due opzioni: • Opzione 1 (next epic): trasferimento interattivo di un file CSV (Comma-Separated Value) EBCDIC su Amazon S3 e interroga il file da Amazon Athena. • Opzione 2: trasferim ento di un set di dati EBCDIC a lunghezza fissa su Amazon S3 come processo batch.	z/OS z/OS Amministratore di mainframe, sviluppatore mainframe

Opzione 1 – Trasferimento interattivo di dati da USS ad Amazon S3 da una sessione USS

Operazione	Description	Competenze richieste
Scarica e trasferisci il file CSV di esempio.	1. Scaricalo `sales-records.csv` dalla sezione Allegati. Questo file fornisce un file CSV di esempio con i record di vendita. 2. Trasferisci il file su z/OS USS. 3. Verifica che il `/u/cliuser/sales-records.csv` file sia leggibile in formato EBCDIC in USS utilizzando un editor di testo a tua scelta.	Sviluppatore di app, sviluppatore di mainframe z/OS

Crea un bucket S3 e carica il file CSV.	1. Crea un bucket S3 per archiviare il file CSV.
	`aws s3 mb s3://<s3_bucket_name>`
	`<s3_bucket_name>` dov'è il nome univoco di un bucket; ad esempio:
	`aws s3 mb s3://DOC-EXAMPLE-BUCKET1`
	2. Carica il file CSV da z/OS USS al bucket S3:
		Sviluppatore di app, sviluppatore di mainframe z/OS

Operazione	Description	Competenze richieste

	`aws s3 cp <csv_file _path> s3://<s3_ bucket_name>`
	
	Esempio:
	
	`aws s3 cp /u/cliuser/ sales-records.csv s3://DOC-EXAMPLE-B UCKET1`
	
	3. Elenca il contenuto del bucket S3 e conferma che includa il file caricato:
	
	`aws s3 ls s3://<s3_ bucket_name>`
	
	Esempio:
	
	`aws s3 ls s3://DOC-EXAMPLE-BUCKET1`	

Operazione	Description	Competenze richieste
Visualizza il bucket S3 e il file caricato.	1. Accedi Console di gestione AWS e apri la console Amazon S3. 2. Naviga per vedere il nuovo bucket S3 e l'oggetto caricato. Per ulteriori informazioni sul caricamento di oggetti, consulta la sezione Guida introduttiva ad Amazon S3 nella documentazione di Amazon S3.	Informazioni generali su AWS

Operazione	Description	Competenze richieste

Esegui una query SQL su una tabella Amazon Athena.	1. Aprire la console Amazon Athena.
	2. Crea una nuova tabella (ad esempio`DOC-EXAMPLE-BUCKET`) utilizzando i dati CSV di Amazon S3. Per ulteriori informazioni, consulta la sezione Interrogazione dell'inventario Amazon S3 con Amazon Athena nella documentazione di Amazon S3.
	3. Esegui la `SELECT` query sulla tabella per visualizzare i dati.
	`SELECT * FROM <table_name>;`
	Esempio:
	`SELECT * FROM DOC-EXAMPLE-BUCKET;`
	L'output della query SQL mostrerà il contenuto del file CSV.
		General AWS, sviluppatore di app

Opzione 2 – Trasferimento di dati da USS ad Amazon S3 utilizzando batch JCL

Operazione	Description	Competenze richieste
Carica il file di esempio.	1. Scaricalo `sales-records-fixed.txt` dalla sezione Allegati. Questo è un file di esempio con i dati di vendita. Rinomina il file di testo, ad `USER.DATA.FIXED` esempio in. 2. Trasferisci il file in un z/OS set di dati a blocco fisso (FB), con lunghezza di 256 record (LRECL) e sequenziale fisico (PS). 3. Utilizzate l'utilità di elenco dei set di dati per verificare che il set di dati sia leggibile in formato EBCDIC nell'`USER.DATA.FIXED` opzione ISPF 3.4. Vedi la sezione Informazioni aggiuntive per un esempio di output.	Sviluppatore di mainframe z/OS
Crea JCL in batch.	Codifica il batch JCL come segue per creare il bucket S3 di destinazione, caricare il set di dati ed elencare il contenuto del bucket. Assicurati di sostituire il nome della directory, i nomi dei file e il nome del bucket con i tuoi valori.	Sviluppatore di mainframe z/OS

Operazione	Description	Competenze richieste

	`//AWSCLICP JOB`
	` ACTINFO1,'IBMUSER'`
	`,CLASS=A,MSGCLASS=`
	`H,MSGLEVEL=(1,1),`
	`// NOTIFY=&SYSUID,TIM`
	`E=1440`
	`//*--------------`
	`------------------`
	`------------------`
	`-------`
	`//* Sample job for AWS`
	` CLI`
	`//*--------------`
	`------------------`
	`------------------`
	`-------`
	`//USSCMD EXEC PGM=BPXBA`
	`TCH`
	`//STDERR  DD SYSOUT=*`
	`//STDOUT  DD SYSOUT=*`
	`//STDENV  DD *`
	` export PATH=/u/c`
	`liuser/python/bin:`
	`$PATH`
	`//STDPARM DD *`
	`SH`
	` export _BPXK_AUT`
	`OCVT=ON;`
	` aws s3 mb s3://DOC-`
	`EXAMPLE-BUCKET2;`
	` cp "//'USER.DATA.FIXE`
	`D'" /tmp/tmpfile;`
	` aws s3 cp /tmp/tmpf`
	`ile s3://DOC-EXAMPLE-`
	`BUCKET2/USER.DATA.F`
	`IXED;`
	` rm /tmp/tmpfile;`
	` aws s3 ls s3://DOC-`
	`EXAMPLE-BUCKET2;`
	`/*`	

Operazione	Description	Competenze richieste
Invia il lavoro JCL in batch.	1. Invia il lavoro JCL che hai codificato nel passaggio precedente. 2. Controlla lo stato del lavoro in System Display and Search Facility (SDSF). In caso di successo, il processo dovrebbe terminare con il codice restituito 0. 3. L'output standard (STDOUT) del job log mostra l'avvenuta creazione del bucket, il caricamento del set di dati e l'elenco dei contenuti del bucket. Per un'illustrazione di esempio della schermata, consultate la sezione Informazioni aggiuntive.	Sviluppatore di mainframe z/OS
Visualizza il set di dati caricato nel bucket S3.	1. Accedi Console di gestione AWS e apri la console Amazon S3. 2. Vai a vedere il file caricato nel bucket di test. 3. Puoi elaborare ulteriormente il `USER.DATA.FIXED` file o analizzarlo utilizzando servizi di analisi come Amazon Redshift.	Informazioni generali su AWS

Risorse correlate

- AWS CLI documentazione della versione 1
- Modernizzazione del mainframe AWS Riferimento ai comandi CLI
- Modernizzazione del mainframe AWS

Informazioni aggiuntive

USER.DATA.FIXED nell'opzione ISPF 3.4 (utilità per l'elenco dei set di dati)

SYSOUT del processo batch inviato

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Esegui il backup e l'archiviazione dei dati del mainframe su Amazon S3 utilizzando BMC AMI Cloud Data

Santosh Kumar Singh, Gilberto Biondo e Maggie Li, Amazon Web Services

Mikhael Liberman, software mainframe Model9

Riepilogo

Questo modello dimostra come eseguire il backup e l'archiviazione dei dati del mainframe direttamente su Amazon Simple Storage Service (Amazon S3), quindi richiamare e ripristinare tali dati sul mainframe utilizzando BMC AMI Cloud Data (precedentemente noto come Model9 Manager). Se stai cercando un modo per modernizzare la tua soluzione di backup e archiviazione nell'ambito di un progetto di modernizzazione del mainframe o per soddisfare i requisiti di conformità, questo modello può aiutarti a raggiungere questi obiettivi.

In genere, le organizzazioni che eseguono applicazioni aziendali principali su mainframe utilizzano una libreria a nastro virtuale (VTL) per eseguire il backup di archivi di dati come file e log. Questo metodo può essere costoso perché utilizza MIPS fatturabili e i dati archiviati su nastri all'esterno del mainframe sono inaccessibili. Per evitare questi problemi, è possibile utilizzare BMC AMI Cloud Data per trasferire in modo rapido ed economico i dati operativi e storici del mainframe direttamente su Amazon S3. È possibile utilizzare BMC AMI Cloud Data per eseguire il backup e l'archiviazione dei dati sfruttando TCP/IP al AWS contempo i motori IBM z Integrated Information Processor (zIIP) per ridurre costi, parallelismo e tempi di trasferimento.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- BMC AMI Cloud Data con una chiave di licenza valida
- Connettività TCP/IP tra il mainframe e AWS
- Un ruolo AWS Identity and Access Management (IAM) per l' read/write accesso a un bucket S3
- Accesso ai prodotti di sicurezza mainframe (RACF) per eseguire i processi BMC AMI Cloud
- Un z/OS agente BMC AMI Cloud (Java versione 8 a 64 bit SR5 FP16 o successiva) con porte di rete disponibili, regole firewall che consentono l'accesso ai bucket S3 e un file system dedicato z/FS
- Requisiti soddisfatti per il server di gestione BMC AMI Cloud

Limitazioni

- BMC AMI Cloud Data archivia i dati operativi in un database PostgreSQL che viene eseguito come contenitore Docker sulla stessa istanza Amazon Elastic Compute Cloud (Amazon) del server di gestione. EC2 Amazon Relational Database Service (Amazon RDS) non è attualmente supportato come backend per BMC AMI Cloud Data. Per ulteriori informazioni sugli ultimi aggiornamenti del prodotto, consulta What's New? nella documentazione BMC.
- Questo modello esegue il backup e l'archiviazione solo dei dati z/OS del mainframe. BMC AMI Cloud Data esegue il backup e l'archiviazione solo dei file mainframe.
- Questo modello non converte i dati in formati aperti standard come JSON o CSV. Utilizza un servizio di trasformazione aggiuntivo come BMC AMI Cloud Analytics (precedentemente noto come Model9 Gravity) per convertire i dati in formati aperti standard. Le applicazioni native del cloud e gli strumenti di analisi dei dati possono accedere ai dati dopo che sono stati scritti nel cloud.

Versioni del prodotto

- BMC AMI Cloud Data versione 2.x

Architecture

Stack tecnologico di origine

- Mainframe con z/OS
- File mainframe come set di dati e file z/OS UNIX System Services (USS)
- Disco mainframe, ad esempio un dispositivo di archiviazione ad accesso diretto (DASD)
- Nastro mainframe (libreria di nastri virtuale o fisica)

Stack tecnologico Target

- Simple Storage Service (Amazon S3)
- EC2 Istanza Amazon in un cloud privato virtuale (VPC)
- AWS Direct Connect
- Amazon Elastic File System (Amazon EFS)

Architettura Target

Il diagramma seguente mostra un'architettura di riferimento in cui gli agenti software BMC AMI Cloud Data su un mainframe gestiscono i processi di backup e archiviazione dei dati legacy che archiviano i dati in Amazon S3.

Il diagramma mostra il flusso di lavoro seguente:

1. Gli agenti software BMC AMI Cloud Data vengono eseguiti su partizioni logiche mainframe (). LPARs Gli agenti software leggono e scrivono i dati del mainframe da DASD o nastro direttamente su Amazon S3 tramite TCP/IP.
2. AWS Direct Connect configura una connessione fisica e isolata tra la rete locale e. AWS Per una maggiore sicurezza, utilizza anche una site-to-site VPN Direct Connect per crittografare i dati in transito.
3. Il bucket S3 archivia i file mainframe come dati di storage di oggetti e gli agenti BMC AMI Cloud Data comunicano direttamente con i bucket S3. I certificati vengono utilizzati per la crittografia HTTPS di tutte le comunicazioni tra l'agente e Amazon S3. La crittografia dei dati di Amazon S3 viene utilizzata per crittografare e proteggere i dati archiviati.
4. I server di gestione BMC AMI Cloud Data funzionano come contenitori Docker sulle EC2 istanze. Le istanze comunicano con gli agenti in esecuzione su mainframe LPARs e bucket S3.
5. Amazon EFS è montato su EC2 istanze attive e passive per condividere lo storage Network File System (NFS). Questo serve a garantire che i metadati relativi a una policy creata sul server di gestione non vadano persi in caso di failover. In caso di failover da parte del server attivo, è possibile accedere al server passivo senza alcuna perdita di dati. In caso di guasto del server passivo, è possibile accedere al server attivo senza alcuna perdita di dati.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon Elastic File System (Amazon EFS) ti aiuta a creare e configurare file system condivisi in Cloud AWS.
- Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è un servizio di storage di oggetti basato sul cloud che consente di archiviare, proteggere e recuperare quasi ogni quantità di dati.

- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.
- AWS Direct Connectcollega la rete interna a una AWS Direct Connect posizione tramite un cavo Ethernet standard in fibra ottica. Con questa connessione, è possibile creare interfacce virtuali direttamente verso i AWS servizi pubblici ignorando i provider di servizi Internet nel percorso di rete.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.

Strumenti BMC

- Il server di gestione BMC AMI Cloud è un'applicazione GUI che viene eseguita come contenitore Docker su Amazon Linux Amazon Machine Image (AMI) per Amazon. EC2 Il server di gestione offre la funzionalità per gestire le attività di BMC AMI Cloud come la creazione di report, la creazione e la gestione di policy, l'esecuzione di archivi e l'esecuzione di backup, richiami e ripristini.
- L'agente BMC AMI Cloud viene eseguito su un mainframe LPAR locale che legge e scrive i file direttamente nell'archivio di oggetti utilizzando TCP/IP. Un'attività avviata viene eseguita su un mainframe LPAR ed è responsabile della lettura e della scrittura dei dati di backup e archiviazione da e verso Amazon S3.
- BMC AMI Cloud Mainframe Command Line Interface (M9CLI) fornisce una serie di comandi per eseguire azioni BMC AMI Cloud direttamente da TSO/E o in batch, senza la dipendenza dal server di gestione.

Epiche

Crea un bucket S3 e una policy IAM

Operazione	Description	Competenze richieste
Crea un bucket S3.	Crea un bucket S3 per archiviare i file e i volumi di cui desideri eseguire il	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	backup e l'archiviazione dal tuo ambiente mainframe.	

Operazione	Description	Competenze richieste
Creare una policy IAM	Tutti i server e gli agenti di gestione BMC AMI Cloud richiedono l'accesso al bucket S3 creato nel passaggio precedente. Per concedere l'accesso richiesto, crea la seguente policy IAM: `{ "Version": "2012-10-17", "Statement": [ { "Sid": "Listfolder", "Action": [ "s3:ListBucket", "s3:GetBucketLocation", "s3:ListBucketVersions" ], "Effect": "Allow", "Resource": [ "arn:aws:s3:::<BucketName>" ] }, { "Sid": "Objectaccess",`	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	"Effect": "Allow", "Action": ["s3:PutObject", "s3:GetObjectAcl", "s3:GetObject", "s3:DeleteObjectVersion", "s3:DeleteObject", "s3:PutObjectAcl", "s3:GetObjectVersion"], "Resource": ["arn:aws:s3:::<BucketName>/*"] }] }	

Ottieni la licenza del software BMC AMI Cloud e scarica il software

Operazione	Description	Competenze richieste
Ottieni una licenza software BMC AMI Cloud.	Per ottenere una chiave di licenza software, contatta il team BMC AMI Cloud. L'output del z/OS `D M=CPU`	Costruisci piombo

Operazione	Description	Competenze richieste
	comando è necessario per generare una licenza.	
Scarica il software BMC AMI Cloud e il codice di licenza.	Ottieni i file di installazione e la chiave di licenza seguendo le istruzioni nella documenta zione BMC.	Amministratore dell'infr astruttura mainframe

Installare l'agente software BMC AMI Cloud sul mainframe

Operazione	Description	Competenze richieste
Installare l'agente software BMC AMI Cloud.	1. Prima di iniziare il processo di installazione, verifica che siano soddisfatti i requisiti software e hardware minimi per l'agente. 2. Per installare l'agente, seguire le istruzioni nella documentazione BMC. 3. Dopo che l'agente ha iniziato a funzionare sul mainframe LPAR, controllate la presenza del `ZM91000I MODEL9 BACKUP AGENT INITIALIZED` messaggio nello spool. Verifica che la connettività sia stata stabilita correttamente tra l'agente e il bucket S3 cercando il `Object store connectivity has been establish`	Amministratore dell'infr astruttura mainframe

Operazione	Description	Competenze richieste
	`ed successfully` messaggio nello STDOUT dell'agente.	

Configurare un server di gestione BMC AMI Cloud su un'istanza EC2

Operazione	Description	Competenze richieste
Crea istanze Amazon EC2 Linux 2.	Avvia due istanze di Amazon EC2 Linux 2 in diverse zone di disponibilità seguendo le istruzioni del Passaggio 1: Avvia un'istanza nella EC2 documentazione di Amazon. L'istanza deve soddisfare i seguenti requisiti hardware e software consigliati: • CPU: minimo 4 core • RAM: minimo 8 GB • Unità: 40 GB • EC2 Istanza consigliata: C5.xlarge • Sistema operativo: Linux • Software: Docker, unzip, VI/vim • Larghezza di banda di rete: minimo 1 GB Per ulteriori informazioni, consulta la documentazione BMC.	Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste
Crea un file system Amazon EFS.	Crea un file system Amazon EFS seguendo le istruzioni della Fase 1: Crea il tuo file system Amazon EFS nella documentazione di Amazon EFS. Durante la creazione del file system, procedi come segue: • Scegli la classe di archiviazione Standard. • Scegli lo stesso VPC che hai usato per avviare le tue EC2 istanze.	Amministratore del cloud, architetto del cloud

Operazione	Description	Competenze richieste

Installa Docker e configura il server di gestione.	Connect alle tue EC2 istanze:
	
	Connettiti alle tue EC2 istanze seguendo le istruzioni di Connect to your Linux istance nella EC2 documentazione di Amazon.
	
	Configura le tue EC2 istanze:
	
	Per ogni EC2 istanza, procedi come segue:
	
	1. Per installare Docker, esegui il comando:
	
	`sudo yum install docker`
	
	2. Per avviare Docker, esegui il comando:
	
	`sudo service docker start`
	
	3. Per convalidare lo stato di Docker, esegui il comando:
	
	`sudo service docker status`
	
	4. Nella `/etc/selinux` cartella, modifica il `config` file in. `SELINUX=permissive`
	
	5. Carica i `VerificationScripts.zip` file
		Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste

	`model9-v2.x.y_build_build-id-server.zip` and (che hai scaricato in precedenza) in una cartella temporanea in una delle EC2 istanze (ad esempio, nella `/var/tmp` cartella dell'istanza).
	6. Per accedere alla `tmp` cartella, esegui il comando:
	`cd/var/tmp`
	7. Per decomprimere lo script di verifica, esegui il comando:
	`unzip VerificationScripts.zip`
	8. Per cambiare la directory, esegui il comando:
	`cd /var/tmp/sysutils/PrereqsScripts`
	9. Per eseguire lo script di verifica, esegui il comando:
	`./M9VerifyPrereqs.sh`
	10. Dopo che lo script di verifica richiede l'input, inserisci l'URL e il numero di porta di Amazon S3. Quindi,	

Operazione	Description	Competenze richieste
	inserisci l' z/OS IP/DNS e il numero di porta. **Note** Lo script esegue un controllo per confermare che l' EC2 istanza può connetter si al bucket S3 e all'agente in esecuzione sul mainframe. Se viene stabilita una connessione, viene visualizzato un messaggio di successo.	

Operazione	Description	Competenze richieste

Installa il software del server di gestione.	1. Create una cartella e una sottocartella nella directory principale (ad esempio `/data/model9`) nell' EC2 istanza in cui intendete rendere attivo il server.
	2. Per installare il `amazon-efs-utils` pacchetto e montare il file system Amazon EFS creato in precedenza, esegui i seguenti comandi:
	`sudo yum install -y amazon-efs-utils`
	`sudo mount -t efs -o tls <File System ID>:/ /data/model9`
	3. Per aggiornare il `/etc/fstab` file dell' EC2 istanza con una voce per il file system Amazon EFS (in modo che Amazon EFS venga rimontato automaticamente al EC2 riavvio di Amazon), esegui il comando:
	`<Amazon-EFS-file-system-id>:/ /data/model9 efs defaults,_netdev 0 0`
	4. Per definire il percorso dei file di installazione di BMC
		Architetto del cloud, amministratore del cloud

Operazione	Description	Competenze richieste

	AMI Cloud e la posizione di installazione di destinazione, eseguire i seguenti comandi per esportare le variabili:
	`export MODEL9_HOME=/data/model9`
	`export M9INSTALL=/var/tmp`
	Note
	Ti consigliamo di aggiungere questi comandi EXPORT allo `.bashrc` script.
	5. Per modificare la directory, esegui il `cd $MODEL9_HOME` comando, quindi crea un'altra sottodirectory eseguendo il `mkdir diag` comando.
	6. Per decomprimere il file di installazione, esegui il comando:
	`unzip $M9INSTALL/model9-<v2.x.y>_build_<build-id>-server.zip`	

Operazione	Description	Competenze richieste

	Note
	Sostituisci `x.y` (la versione) e `build-id` con i tuoi valori.
	7. Per distribuire l'applica zione, esegui i seguenti comandi:
	`docker load -i $MODEL9_HOME/model9-<v2.x.y>_build_<build-id>.docker`
	`docker load -i $MODEL9_HOME/postgres-12.10-x86.docker.gz`
	Note
	Sostituisci `v2.x.y` (la versione) e `build-id` con i tuoi valori.
	8. Nella `$MODEL9_HOME/conf` cartella, aggiorna il `model9-local.yml` file.
	Note
	Alcuni parametri hanno valori predefiniti e altri possono essere	

Operazione	Description	Competenze richieste

	aggiornati se necessario. Per ulteriori informazioni, consultate le istruzioni contenute nel `model9-local.yml` file.
	9. Create un file chiamato **`$MODEL9_HOME/conf`** , quindi aggiungete i seguenti parametri al file:
	`TZ=America/New_York`
	`EXTRA_JVM_ARGS=-Xmx2048m`
	10. Per creare un bridge di rete Docker, esegui il comando:
	`docker network create -d bridge model9network`
	11. Per avviare il contenitore del database PostgreSQL per BMC AMI Cloud, eseguire il seguente comando:
	`docker run -p 127.0.0.1:5432:5432 \`
	`-v $MODEL9_HOME/db/data:/var/lib/postgresql/data:z \`	

Operazione	Description	Competenze richieste
	`--name model9db --restart unless-stopped \ --network model9network \ -e POSTGRES_PASSWORD=model9 -e POSTGRES_DB=model9 -d postgres:12.10`	
	12. Dopo l'avvio dell'esecuzione del contenitore PostgreSQL, esegui il comando seguente per avviare il server delle applicazioni: `docker run -d -p 0.0.0.0:443:443 -p 0.0.0.0:80:80 \ --sysctl net.ipv4.tcp_keepalive_time=600 \ --sysctl net.ipv4.tcp_keepalive_intvl=30 \ --sysctl net.ipv4.tcp_keepalive_probes=10 \ -v $MODEL9_HOME:/model9:z -h $(hostname) --restart unless-stopped \ --env-file $MODEL9_HOME/conf/model9.env \ --network model9network \ --name model9-v2.x.y model9:<v2.x.y>.<build-id>`	

Operazione	Description	Competenze richieste
	Note Sostituisci `v2.x.y` (la versione) e `build-id` con i tuoi valori. 13. Per verificare lo stato di salute di entrambi i contenitori, esegui il comando: `docker ps -a` 14. Per installare un server di gestione sulle EC2 istanze passive, ripeti i passaggi 1—4, 7 e 10—13. **Note** Per risolvere i problemi, accedi ai log archiviati nella cartella. `/data/model9/logs/` Per ulteriori informazioni, consulta la documentazione BMC.	

Aggiungere un agente e definire una politica di backup o archiviazione sul server di gestione BMC AMI Cloud

Operazione	Description	Competenze richieste
Aggiungi un nuovo agente.	Prima di aggiungere un nuovo agente, conferma quanto segue: • Un agente BMC AMI Cloud è in esecuzione sul mainframe LPAR ed è stato inizializzato completamente. Identifica l'agente cercando il messaggio di `ZM91000I MODEL9 BACKUP AGENT INITIALIZED` inizializzazione nello spool. • Un contenitore Docker per il server di gestione è completamente inizializzato e funzionante. È necessario creare un agente sul server di gestione prima di definire qualsiasi politica di backup e archiviazione. Per creare l'agente, effettuate le seguenti operazioni: 1. Utilizza un browser Web per accedere al server di gestione distribuito sulla tua EC2 macchina Amazon, quindi accedi con le tue credenziali mainframe.	Amministratore o sviluppatore dello storage mainframe

Operazione	Description	Competenze richieste
	2. Scegli la scheda AGENTI, quindi scegli AGGIUNGI NUOVO AGENTE. 3. In Nome, inserisci il nome dell'agente. 4. Per Nome host/Indirizzo IP, inserite il nome host o l'indirizzo IP del mainframe. 5. Per Port, inserisci il tuo numero di porta. 6. Scegli TEST CONNECTIO N. Se la connettività è stata stabilita correttam ente, viene visualizzato un messaggio di successo. 7. Scegli CREATE. Dopo aver creato l'agente, vedrai lo stato della connessio ne rispetto all'object storage e all'agente mainframe in una nuova finestra che appare nella tabella.	

Operazione	Description	Competenze richieste
Crea una politica di backup o archiviazione.	1. Scegli POLICY. 2. Scegli CREA POLITICA. 3. Nella pagina CREA UNA NUOVA POLITICA, inserisci le specifiche della tua politica. **Note** Per ulteriori informazioni sulle specifiche disponibili, vedere Creazione di una nuova politica nella documentazione BMC. 4. Scegli Fine. 5. La nuova politica è ora elencata sotto forma di tabella. Per visualizzare questa tabella, scegli la scheda POLITICHE.	Amministratore o sviluppatore dello storage mainframe

Esegui la politica di backup o archiviazione dal server di gestione

Operazione	Description	Competenze richieste
Esegui la politica di backup o archiviazione.	Esegui la politica di backup o archiviazione dei dati creata in precedenza dal server di gestione manualmente o automaticamente (in base	Amministratore o sviluppatore dello storage mainframe

Operazione	Description	Competenze richieste
	a una pianificazione). Per eseguire la policy manualmente: 1. Scegli la scheda POLITICHE dal menu di navigazione. 2. Sul lato destro della tabella relativa alla policy che desideri eseguire, scegli il menu a tre punti. 3. Scegli Esegui ora. 4. Nella finestra pop-up di conferma, scegli SÌ, ESEGUI POLICY NOW. 5. Dopo l'esecuzione della policy, verifica lo stato di esecuzione nella sezione relativa all'attività della policy. 6. Per la policy in esecuzione, scegli il menu a tre punti, quindi scegli Visualizza registro di esecuzione per visualizzare i log. 7. Per verificare che il backup sia stato creato, controlla il bucket S3.	

Operazione	Description	Competenze richieste
Ripristina la politica di backup o archiviazione.	1. Nel menu di navigazione, scegli la scheda POLITICHE. 2. Scegli la politica su cui eseguire il processo di ripristino. Verranno elencate tutte le attività di backup o archiviazione eseguite in passato in base a quella specifica policy. 3. Per selezionare i backup che desideri ripristinare, scegli la colonna Data-ora. Il nome `file/Volume/Storage` del gruppo mostra i dettagli di esecuzione della politica. 4. Sul lato destro della tabella, scegli il menu a tre punti, quindi scegli RIPRISTINA. 5. Nella finestra pop-up, inserisci il nome, il volume e il gruppo di archiviazione del target, quindi scegli RESTORE. 6. Inserisci le credenziali del mainframe, quindi scegli nuovamente RESTORE. 7. Per verificare che il ripristino sia andato a buon fine, controlla i log o il mainframe.	Amministratore o sviluppatore dello storage mainframe

Esegui la policy di backup o archiviazione dal mainframe

Operazione	Description	Competenze richieste
Esegui la politica di backup o archiviazione utilizzando M9CLI.	Utilizza M9CLI per eseguire processi di backup e ripristino da TSO/E, REXX o tramite JCLs senza configurare regole sul server di gestione BMC AMI Cloud. Utilizzando TSO/E: Se usi TSO/E, assicurati che sia concatenato a. `M9CLI REXX TSO` Per eseguire il backup di un set di dati tramite TSO/E, usa il comando. `TSO M9CLI BACKDSN <DSNAME>` **Note** Per ulteriori informazioni sui comandi M9CLI, vedere il riferimento CLI nella documentazione BMC. Utilizzando: JCLs Per eseguire la politica di backup e archiviazione utilizzando JCLs, esegui il `M9CLI` comando. Utilizzo delle operazioni in batch:	Amministratore o sviluppatore dello storage mainframe

Operazione	Description	Competenze richieste

	L'esempio seguente mostra come archiviare un set di dati eseguendo il `M9CLI` comando in batch:
	
	`//JOBNAME JOB …`
	`//M9CLI EXEC PGM=IKJEF`
	`T01`
	`//STEPLIB DD DISP=SHR,`
	`DSN=<MODEL9 LOADLIB>`
	`//SYSEXEC DD DISP=SHR,`
	`DSN=<MODEL9 EXEC LIB>`
	`//SYSTSPRT DD SYSOUT=*`
	`//SYSPRINT DD SYSOUT=*`
	`//SYSTSIN DD TSO M9CLI`
	` ARCHIVE`
	` M9CLI ARCHIVE <DSNNAME`
	` OR DSN PATTERN>`
	`/`	

Operazione	Description	Competenze richieste
Esegui la politica di backup o archiviazione nel batch JCL.	BMC AMI Cloud fornisce una routine JCL di esempio chiamata M9SAPIJ. È possibile personalizzare M9SAPIJ per eseguire una politica specifica creata sul server di gestione con un JCL. Questo processo può anche far parte di un programma di pianificazione in batch per l'esecuzione automatica dei processi di backup e ripristino. Il processo batch prevede i seguenti valori obbligatori: • Nome IP address/host del server di gestione • Numero della porta • ID o nome della policy (creato sul server di gestione) **Note** È inoltre possibile modificare altri valori seguendo le istruzioni sul job di esempio.	Amministratore o sviluppatore dello storage mainframe

Risorse correlate

- Modernizzazione del mainframe con AWS (documentazione AWS)

- In che modo il backup su cloud per mainframe riduce i costi con Model9 e AWS (blog di AWS Partner Network)
- Come abilitare l'analisi dei dati mainframe su AWS utilizzando Model9 (AWS Partner Network Blog)
- Raccomandazioni sulla resilienza di AWS Direct Connect (documentazione AWS)
- Documentazione BMC AMI Cloud (sito web BMC)

Crea programmi COBOL Db2 usando e Modernizzazione del mainframe AWS AWS CodeBuild

Luis Gustavo Dantas e Eduardo Zimelewicz, Amazon Web Services

Riepilogo

> **Note**
>
> Modernizzazione del mainframe AWS Service (Managed Runtime Environment experience) non è più aperto a nuovi clienti. Per funzionalità simili a Modernizzazione del mainframe AWS Service (Managed Runtime Environment experience), esplora Modernizzazione del mainframe AWS Service (Self-Managed Experience). I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta Modifica della Modernizzazione del mainframe AWS disponibilità.

Questo modello spiega come creare un semplice AWS CodeBuild progetto per precompilare e associare i programmi COBOL Db2 utilizzando gli strumenti Replatform. Modernizzazione del mainframe AWS Ciò consente la distribuzione e l'esecuzione di questi programmi nell'ambiente di runtime Replatform. Modernizzazione del mainframe AWS

COBOL, un linguaggio di programmazione orientato al business, alimenta molte applicazioni critiche grazie alla sua affidabilità e leggibilità. IBM Db2, un sistema di gestione di database relazionali, gestisce grandi volumi di dati in modo efficiente e si integra con i programmi COBOL tramite SQL. Insieme, COBOL e Db2 costituiscono la spina dorsale delle operazioni cruciali in settori come quello finanziario e governativo, nonostante l'emergere di nuove tecnologie.

La migrazione dei componenti COBOL e Db2 dall'ambiente mainframe ad altre piattaforme comporta sfide quali la compatibilità della piattaforma, la complessità dell'integrazione, la migrazione dei dati e l'ottimizzazione delle prestazioni. Lo spostamento di questi componenti critici richiede un'attenta pianificazione, competenze tecniche e risorse per garantire una migrazione senza intoppi pur mantenendo affidabilità e funzionalità.

Il Modernizzazione del mainframe AWS servizio fornisce strumenti e risorse per ripiattaforma applicazioni e database mainframe da eseguire sull' AWS infrastruttura, come le istanze Amazon Elastic Compute Cloud (Amazon EC2). Ciò comporta lo spostamento dei carichi di lavoro mainframe sul cloud senza importanti modifiche al codice.

Il processo di precompilazione e associazione di Db2 è essenziale per ottimizzare le prestazioni e l'affidabilità delle applicazioni di database. La precompilazione trasforma le istruzioni SQL incorporate in codice eseguibile, riducendo il sovraccarico di runtime e migliorando l'efficienza. Il processo di associazione collega il codice precompilato con le strutture del database, facilitando i percorsi di accesso e l'ottimizzazione delle query. Questo processo garantisce l'integrità dei dati, migliora la reattività delle applicazioni e protegge dalle vulnerabilità di sicurezza. Le applicazioni correttamente precompilate e associate riducono al minimo il consumo di risorse, migliorano la scalabilità e mitigano i rischi degli attacchi di SQL injection.

Prerequisiti e limitazioni

Prerequisiti

- Un accesso Account AWS alla console a livello amministrativo.
- Un sistema di database IBM Db2, ad esempio IBM Db2 for z/OS o Db2 per Linux, Unix e Windows (LUW).
- Il software IBM Data Server Client, disponibile per il download dal sito Web IBM. Per ulteriori informazioni, consulta i tipi di client IBM Data Server e Driver Data Server.
- Un programma COBOL Db2 da compilare e associare. In alternativa, questo modello fornisce un programma di esempio di base che è possibile utilizzare.
- Un cloud privato virtuale (VPC) AWS con una rete privata. Per informazioni sulla creazione di un VPC, consulta la documentazione di Amazon Virtual Private Cloud (Amazon VPC).
- Un repository per il controllo del codice sorgente come o. GitHub GitLab

Limitazioni

- Per le AWS CodeBuild quote, vedere Quote per. AWS CodeBuild
- Alcune Servizi AWS non sono disponibili in tutte. Regioni AWS Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Architecture

Stack tecnologico di origine

Lo stack di origine include:

- Programmi COBOL che utilizzano un database Db2 per archiviare i dati
- Compilatore IBM COBOL e Db2 per precompilatore z/OS
- Altre parti della configurazione del mainframe, come il file system, il gestore delle transazioni e lo spool

Stack tecnologico Target

L'approccio di questo modello funziona per due opzioni: spostare i dati da Db2 for z/OS a Db2 per LUW o rimanere su Db2 per z/OS. L'architettura di destinazione include:

- Programmi COBOL che utilizzano un database Db2 per archiviare i dati
- Modernizzazione del mainframe AWS Strumenti di compilazione per ripiattaforme
- AWS CodeBuild come infrastruttura per creare l'applicazione
- Altre Cloud AWS risorse come Amazon Linux

Architettura Target

Il diagramma illustra quanto segue:

1. L'utente carica il proprio codice in un repository di controllo del codice sorgente come o. GitHub GitLab
2. AWS CodePipeline rileva la modifica e ottiene il codice dal repository.
3. CodePipeline avvia AWS CodeBuild e invia il codice.
4. CodeBuild segue le istruzioni contenute nel `buildspec.yml` modello (fornito nella sezione Informazioni aggiuntive) per:
   a. Ottieni l'IBM Data Server Client da un bucket Amazon Simple Storage Service (Amazon S3).
   b. Installa e configura IBM Data Server Client.
   c. Recupera le credenziali Db2 da. Gestione dei segreti AWS
   d. Connettiti al server Db2.
   e. Precompila, compila e associa il programma COBOL.
   f. Salva i prodotti finiti in un bucket S3 per utilizzarli. AWS CodeDeploy
5. CodePipeline inizia. CodeDeploy

6. CodeDeploy coordina i suoi agenti, che sono già installati negli ambienti di runtime. Gli agenti recuperano l'applicazione da Amazon S3 e la installano in base alle istruzioni contenute in. `appspec.yml`

Per semplificare le cose e concentrarsi sulla compilazione, le istruzioni riportate in questo schema coprono i passaggi da 1 a 4 ma non includono l'implementazione del programma COBOL Db2.

Automazione e scalabilità

Per semplicità, questo modello descrive come fornire le risorse manualmente. Tuttavia, sono disponibili numerose opzioni di automazione, come CloudFormation, e HashiCorp Terraform AWS Cloud Development Kit (AWS CDK), che automatizzano queste attività. Per ulteriori informazioni, consulta la documentazione CloudFormatione AWS CDK.

Tools (Strumenti)

Servizi AWS

- AWS CodeBuildè un servizio di compilazione completamente gestito che consente di compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS CodeDeployautomatizza le distribuzioni su Amazon EC2 o su istanze AWS Lambda , funzioni o servizi Amazon Elastic Container Service (Amazon ECS) in locale.
- AWS CodePipelineti aiuta a modellare e configurare rapidamente le diverse fasi di un rilascio software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- Modernizzazione del mainframe AWSfornisce strumenti e risorse per aiutarvi a pianificare e implementare la migrazione e la modernizzazione dai mainframe agli ambienti di runtime AWS gestiti.

Altri strumenti

- Immagine Amazon ECR per gli strumenti Modernizzazione del mainframe AWS Replatform. Per compilare un'applicazione COBOL, devi iniziare CodeBuild utilizzando un'immagine Amazon Elastic Container Registry (Amazon ECR) che contiene gli strumenti Replatform: Modernizzazione del mainframe AWS

  `673918848628.dkr.ecr.<your-region>.amazonaws.com/m2-enterprise-build-tools:9.0.7.R1`

Per ulteriori informazioni sull'immagine ECR disponibile, consulta il tutorial nella Guida per l'utente.Modernizzazione del mainframe AWS

- Il software IBM Data Server Client è essenziale per precompilare e associare i programmi COBOL Db2. CodeBuild Funge da ponte tra il compilatore COBOL e Db2.

Best practice

- Non tutti i programmi COBOL si affidano a Db2 come livello di persistenza dei dati. Assicuratevi che le direttive di compilazione per l'accesso a Db2 siano applicate solo ai programmi COBOL progettati specificamente per interagire con Db2. Implementate una logica per distinguere tra i programmi COBOL Db2 e i programmi COBOL che non utilizzano Db2.
- Ti consigliamo di evitare di compilare programmi che non sono stati modificati. Implementa un processo per identificare quali programmi richiedono la compilazione.

Epiche

Crea l'infrastruttura cloud

Operazione	Description	Competenze richieste
Crea un bucket S3 per ospitare il client IBM Data Server e gli artefatti della pipeline.	È necessario configurare un bucket S3 per (a) caricare IBM Data Server Client, (b) archiviare il codice dal repository e (c) archiviar e i risultati del processo di compilazione. 1. Accedi a e apri la console Amazon S3. Console di gestione AWS 2. Scegli un bucket S3 esistente o creane uno nuovo. Nota l'Amazon	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	Resource Name (ARN) del bucket per utilizzi futuri. Per informazioni su come creare un bucket S3, consulta la documentazione di Amazon S3.	
Carica l'IBM Data Server Client nel bucket S3.	1. Sulla console Amazon S3, scegli il bucket per aprirlo. 2. Scegli Crea cartella, specificane il nome come client, quindi scegli Crea cartella. 3. Apri la cartella client, scegli Carica, Aggiungi file. 4. Scegli il file IBM Data Server Client che hai scaricato in precedenza dal sito Web IBM nel tuo file system locale. Il nome del file deve essere simile a o. `v11.5.8_linuxx64_client.tar.gz` `v11.5.9_linuxx64_client.tar.gz` 5. Scegli Apri, Carica e attendi il completamento del caricamento. 6. Nella scheda File e cartelle, scegli Data Server Client e annota il relativo URI S3.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Crea un Gestione dei segreti AWS segreto per le tue credenziali Db2.	Per creare un segreto per archiviare in modo sicuro le tue credenziali: DB2 1. Nella console Secrets Manager, scegli Archivia un nuovo segreto. 2. Nel riquadro Scegli il tipo di segreto, scegli Altro tipo di segreto e Testo normale. 3. Nella casella Testo normale, digita le tue credenziali Db2 utilizzando la seguente struttura JSON. `{ "username": "<your-db2-user-name>", "password": "<your-db2-password>", "db2node": "db2dev", "db2host": "<your-db2-hostname-or-IP>", "db2port": <your-db2-port>, "db2name": "<your-db2-location>", "qualifier": "<your-db2-qualifier>" }`	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	4. Scegli Avanti e assegna al segreto un nome come. `dev-db2-cred` 5. Scegli Avanti, Avanti e Store. Per ulteriori informazioni sulla creazione di segreti, consulta la documentazione di Secrets Manager.	

Operazione	Description	Competenze richieste
Verifica che Db2 sia accessibi le dalla sottorete VPC.	AWS CodeBuild necessita di una connessione al server Db2 in modo che il Data Server Client possa eseguire operazioni di precompilazione e associazione. Assicurati che CodeBuild possa raggiunge re il server Db2 tramite una connessione sicura. 1. Apri la Console Amazon VPC. 2. Nel pannello di navigazione, scegli Sottoreti e annota le IDs sottoreti private su cui funzioneranno. IPv4 CIDRs CodeBuild 3. Aggiorna le attuali impostazi oni di controllo dell'acce sso alla rete per il tuo sistema Db2 introduce ndo una regola in entrata. Questa regola dovrebbe consentire l'accesso TCP personalizzato alla porta Db2 esclusivamente dalla sottorete associata al CIDRs progetto. CodeBuild	Amministratore di rete, General AWS

Crea gli artefatti dell'applicazione

Operazione	Description	Competenze richieste
Create la risorsa COBOL Db2.	1. Se desiderate utilizzare un semplice esempio di COBOL Db2, salvate il seguente codice sorgente con nome. `CDB2SMP.cbl` In alternativa, potete sostituire questo esempio con un programma che già possedete. `IDENTIFICATION DIVISION. PROGRAM-ID. CDB2SMP. DATA DIVISION. WORKING-STORAGE SECTION. 01 WS-NAME PIC X(100). PROCEDURE DIVISION. EXEC SQL SELECT NAME INTO :WS-NAME FROM SYSIBM.SYSTABLES END-EXEC GOBACK.` 2. Conferma le modifiche e invia il file al tuo repository.	Sviluppatore di app
Crea il file `buildspec.yml` .	1. Genera il `buildspec.yml` file in base all'esempio fornito nella	AWS DevOps

Operazione	Description	Competenze richieste
	sezione Informazioni aggiuntive. 2. Conferma le modifiche e invia il file al tuo repository.	
Connect il tuo repository a. CodePipeline	1. Aprire la AWS console Strumenti di sviluppo. 2. Nel pannello di navigazione, scegli Impostazioni, Connessioni. 3. Segui le istruzioni contenute nella documentazione della console Developer Tools del provider di origine di tua scelta. Avrai bisogno dell'Amazon Resource Name (ARN) per la connessione quando crei la policy AWS Identity and Access Management (IAM) per una CodePipeline fase successiva.	AWS DevOps

Configurazione delle autorizzazioni

Operazione	Description	Competenze richieste
Crea una policy IAM per CodeBuild.	Il CodeBuild progetto richiede l'accesso ad alcune risorse, tra cui Secrets Manager e Amazon S3.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	Per configurare le autorizzazioni necessarie: 1. Apri la console IAM. 2. Nel riquadro di navigazione, scegli Policies, Create Policy, quindi seleziona il CodeBuild servizio. 3. Passa la formattazione da Visual a JSON e copia la CodeBuild policy fornita nella sezione Informazioni aggiuntive nel campo dell'editor delle politiche. 4. Assegna un nome a questa politica e salvala per riferimenti futuri nel passaggio successivo. Per ulteriori informazioni sulla creazione di policy IAM, consulta la documentazione IAM.	

Operazione	Description	Competenze richieste
Crea un ruolo IAM per CodeBuild.	Per rendere disponibili le politiche di sicurezza CodeBuild, devi configurare un ruolo IAM. Per creare questo ruolo: 1. Sulla console IAM, nel riquadro di navigazione, scegli Ruoli, Crea ruolo. 3. Per il tipo di entità affidabil e, mantieni l'Servizio AWSimpostazione predefinita. 4. Per Caso d'uso, seleziona il CodeBuild servizio, quindi scegli Avanti. 4. Nell'elenco delle policy IAM disponibili, individua la policy per cui hai creato CodeBuild , quindi scegli Avanti per associarla al ruolo. 5. Specificate un nome per il ruolo e scegliete Crea ruolo per salvarlo per riferimenti futuri in CodeBuild. Per ulteriori informazioni sulla creazione di un ruolo IAM per un Servizio AWS, consulta la documentazione IAM.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Crea una policy IAM per CodePipeline.	La AWS CodePipeline pipeline richiede l'accesso ad alcune risorse, tra cui il repository di codice e Amazon S3. Ripeti i passaggi forniti in precedenza per CodeBuild creare una policy IAM per CodePipeline (nel passaggio 2, scegli CodePipelineinvece di). CodeBuild	AWS DevOps

Operazione	Description	Competenze richieste

Crea un ruolo IAM per CodePipeline.	Per rendere disponibili le politiche di sicurezza CodePipeline, devi configurare un ruolo IAM.
	
	Per creare questo ruolo:
	
	1. Sulla console IAM, scegli Roles, Create Role.
	2. Per Trusted entity type (Tipo di entità attendibile), scegli Custom trust policy (Policy di attendibilità personalizzata).
	
	Verrà visualizzata una policy con un `Principal` elemento vuoto.
	3. Sulla `Principal` linea, tra le parentesi, aggiungi:
	
	`"Service": "codepipeline.amazonaws.com"`
	
	La politica di fiducia sarà simile a questa:
	
	`{ "Version": "2012-10-17", "Statement": [ { "Sid": "Statement1", "Effect": "Allow", "Principal": {`
		AWS DevOps

Operazione	Description	Competenze richieste
	<pre>"Service": "codepipeline.amazonaws.com" }, "Action": "sts:AssumeRole" }] }</pre> 4. Scegli Next (Successivo). 5. Nell'elenco delle policy IAM disponibili, individua la policy per cui hai creato CodePipeline, quindi scegli Avanti per associarla al ruolo. 6. Specificate un nome per il ruolo e scegliete Crea ruolo per salvarlo per riferimenti futuri in CodePipeline.	

Compila e associa il programma COBOL Db2

Operazione	Description	Competenze richieste
Crea una CodePipeline pipeline e un CodeBuild progetto.	Per creare una CodePipeline pipeline e il CodeBuild progetto che compila e associa il programma COBOL Db2: 1. Apri la CodePipeline console e scegli Crea pipeline, Crea pipeline personalizzata.	AWS DevOps

Operazione	Description	Competenze richieste
	2. Specificate un nome per la pipeline. 3. Per Ruolo di servizio, scegli Ruolo di servizio esistente e scegli specifica l'ARN per il ruolo IAM per cui hai creato. CodePipeline 4. Espandi Impostazioni avanzate, scegli Posizione personalizzata, scegli il bucket S3 che hai creato in precedenza, quindi scegli Avanti. 5. Per Provider di origine, seleziona il tuo fornitore di origine di terze parti e fornisci le informazioni pertinenti per il provider:. a. Per Connessione, seleziona la connessione creata per il provider di origine. b. Per Nome del repository, seleziona il tuo repositor y. c. Per Ramo predefinito, selezionate il ramo che memorizza il programma COBOL e. `buildspec .yml` d. Scegli Next (Successi vo).	

Operazione	Description	Competenze richieste
	6. Per Build provider, scegli Altri fornitori di build, AWS CodeBuild. 7. Per Nome del progetto, scegli Crea progetto. La console visualizza una CodeBuild finestra in cui puoi creare un progetto di compilazione. In questa finestra: a. Inserisci un nome per il progetto. b. Per Environment image (Immagine ambiente) , scegli Custom image (Immagine personali zzata). c. Per Tipo di ambiente, scegli Linux Container. d. Per l'account ECR, scegli Altro account ECR. e. Per l'URI del repositor y Amazon ECR, inserisci:. `673918848628.dkr.ecr.<your-region>.amazonaws.com/m2-enterprise-build-tools:8.0.9.R1` f. Per Ruolo di servizio, scegli Ruolo di servizio	

Operazione	Description	Competenze richieste
	esistente e seleziona il ruolo per cui hai creato. CodeBuild g. Espandi la sezione Configurazione aggiuntiva, quindi scegli il VPC, le sottoreti private e il gruppo di sicurezza per questo progetto. h. Nella sezione Buildspec, scegli Usa un file buildspec. i. Alla fine della finestra, scegli Continua a. CodePipeline La CodeBuild finestra si chiuderà così potrai tornare alla CodePipeline console. 8. Tornando alla CodePipeline console, scegli Avanti. 9. Nel riquadro Aggiungi fase di distribuzione, scegli Salta fase di distribuzione e conferma. 10. Controlla i parametri della pipeline, quindi scegli Crea pipeline.	
Esamina l'output.	Verifica il successo della build esaminando i log di CodePipeline compilazione.	AWS DevOps

Operazione	Description	Competenze richieste

Controlla i risultati in Db2.	Verifica la versione del pacchetto nella tabella SYSPLAN.
	
	`select CAST(NAME AS VARCHAR(10)) as name, VALIDATE, LAST_BIND_TIME, LASTUSED, CAST(PKGVERSION AS VARCHAR(10)) as PKGVERSION from SYSIBM.SYSPLAN where NAME = 'CDB2SMP' order by LAST_BIND_TIME desc`
	
	La versione deve corrispondere all'ID di CodeBuild build, che è CDB2SMP nel nostro esempio:
	
	`NAME VALIDATE LAST_BIND_TIME LASTUSED PKGVERSION ---------- -------- ---------- -------- CDB2SMP B 2024-05-18-11.53.11.503738 01/01/0001 19`	

risoluzione dei problemi

Problema	Soluzione
Occasionalmente, la AWS console cambia regione quando ci si sposta da un servizio all'altro.	Assicurati di verificare la selezione Regione AWS ogni volta che passi da un servizio all'altro . Il Regione AWS selettore si trova nell'angolo superiore destro della finestra della console.
Può essere difficile identificare i problemi di connettività di Db2. CodeBuild	Per risolvere i problemi di connettività, aggiungi il seguente comando DB2 connect al file. `buildspec.yml` Questa aggiunta consente di eseguire il debug e risolvere i problemi di connettività. `db2 connect to $DB_NAME user $DB2USER using $DB2PASS`
A volte, il riquadro dei ruoli nella console IAM non mostra immediatamente la policy IAM che hai creato.	Se riscontri un ritardo, aggiorna lo schermo per visualizzare le informazioni più recenti.

Risorse correlate

Documentazione IBM

- Client e tipi di driver IBM Data Server
- Scarica i tipi di client e driver IBM Data Server

AWS documentazione

- Guida per l'utente di Amazon S3
- AWS CodeBuild Guida per l'utente
- Modernizzazione del mainframe AWS Guida per l'utente

- Gestione dei segreti AWS Guida per l'utente
- AWS CodePipeline Guida per l'utente
- AWS CodeDeploy Guida per l'utente

Informazioni aggiuntive

CodeBuild policy

Sostituisci i segnaposto<RegionID>,, <AccountID> <SubnetARN><BucketARN>, e <DB2CredSecretARN> con i tuoi valori.

```
{"Version": "2012-10-17",
    "Statement": [
        {"Action": "ecr:GetAuthorizationToken", "Effect": "Allow", "Resource": "*" },
        {"Action": ["ecr:GetDownloadUrlForLayer", "ecr:BatchGetImage",
                    "ecr:BatchCheckLayerAvailability"],
         "Effect": "Allow",
         "Resource": "arn:aws:ecr:*:673918848628:repository/m2-enterprise-build-
tools"},
        {"Action": "s3:PutObject", "Effect": "Allow", "Resource": "arn:aws:s3:::aws-m2-
repo-*/*"},
        {"Action": ["logs:PutLogEvents", "logs:CreateLogStream",
 "logs:CreateLogGroup"],
         "Effect": "Allow", "Resource": "arn:aws:logs:<RegionId>:<AccountId>:*"},
        {"Action": ["ec2:DescribeVpcs", "ec2:DescribeSubnets",
                    "ec2:DescribeSecurityGroups", "ec2:DescribeNetworkInterfaces",
                    "ec2:DescribeDhcpOptions", "ec2:DeleteNetworkInterface",
                    "ec2:CreateNetworkInterface"],
         "Effect": "Allow", "Resource": "*"},
        {"Action": "ec2:CreateNetworkInterfacePermission",
         "Effect": "Allow", "Resource": ["<SubnetARN>"]},
        {"Action": "s3:*", "Effect": "Allow", "Resource": ["<BucketARN>/
*","<BucketARN>"]},
        {"Action": "secretsmanager:GetSecretValue",
         "Effect": "Allow", "Resource": "<DB2CredSecretARN>"}
    ]
}
```

CodePipeline policy

Sostituisci i segnaposto <BucketARN> e <ConnectionARN> con i tuoi valori.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {"Action": ["s3:List*", "s3:GetObjectVersion", "s3:GetObject",
 "s3:GetBucketVersioning" ],
        "Effect": "Allow",
        "Resource": ["<BucketARN>/*", "<BucketARN>"]},
        {"Action": ["codebuild:StartBuild", "codebuild:BatchGetBuilds"],
         "Effect": "Allow", "Resource": "*"},
        {"Action": ["codestar-connections:UseConnection"],
         "Effect": "Allow", "Resource": "<ConnectionARN>"}
        ]
}
```

`buildspec.yml`

Sostituisci il <your-bucket-name> segnaposto con il nome effettivo del bucket S3.

```
version: 0.2
phases:
  pre_build:
    commands:
      - /var/microfocuslicensing/bin/mfcesd -no > /var/microfocuslicensing/logs/
mfcesd_startup.log 2>&1 &
      - |
        mkdir $CODEBUILD_SRC_DIR/db2client
        aws s3 cp s3://<your-bucket-name>/v11.5.8_linuxx64_client.tar.gz
 $CODEBUILD_SRC_DIR/db2client/ >> /dev/null 2>&1
        tar -xf $CODEBUILD_SRC_DIR/db2client/v11.5.8_linuxx64_client.tar.gz -C
 $CODEBUILD_SRC_DIR/db2client/
        cd $CODEBUILD_SRC_DIR/db2client/
        ./client/db2_install -f sysreq -y -b /opt/ibm/db2/V11.5 >> /dev/null 2>&1

        useradd db2cli
        /opt/ibm/db2/V11.5/instance/db2icrt -s client -u db2cli db2cli
        DB2CRED=$(aws secretsmanager get-secret-value --secret-id dev-db2-cred | jq -r
 '.SecretString | fromjson')
        read -r DB2USER DB2PASS DB_NODE DB_HOST DB_PORT DB_NAME DB_QUAL <<<$(echo
 $DB2CRED | jq -r
 '.username, .password, .db2node, .db2host, .db2port, .db2name, .qualifier')
        . /home/db2cli/sqllib/db2profile
        db2 catalog tcpip node $DB_NODE remote $DB_HOST server $DB_PORT
        db2 catalog db $DB_NAME as $DB_NAME at node $DB_NODE authentication server
```

```
  build:
    commands:
      - |
        revision=$CODEBUILD_SRC_DIR/loadlib
        mkdir -p $revision; cd $revision
        . /opt/microfocus/EnterpriseDeveloper/bin/cobsetenv
        cob -zU $CODEBUILD_SRC_DIR/CDB2SMP.cbl -C "DB2(DB==${DB_NAME} PASS==${DB2USER}.
${DB2PASS} VERSION==${CODEBUILD_BUILD_NUMBER} COLLECTION==DB2AWSDB"
artifacts:
  files:
    - "**/*"
  base-directory: $revision
```

Create un Micro Focus Enterprise Server PAC con Amazon EC2 Auto Scaling e Systems Manager

Kevin Yung e Krithika Palani Selvam, Amazon Web Services

Peter Woods, Nessuno

Abraham Rondon, Micro Focus

Riepilogo

Questo modello introduce un'architettura scalabile per applicazioni mainframe che utilizza Micro Focus Enterprise Server in Scale-Out Performance and Availability Cluster (PAC) e un gruppo Amazon Elastic Compute Cloud (Amazon) EC2 Auto Scaling su Amazon Web Services ().AWS La soluzione è completamente automatizzata con AWS Systems Manager i ganci del ciclo di vita di Amazon EC2 Auto Scaling. Utilizzando questo modello, è possibile configurare le applicazioni mainframe online e in batch per ottenere un'elevata resilienza grazie alla scalabilità interna e orizzontale automatica in base alle esigenze di capacità.

> **Note**
>
> Questo modello è stato testato con Micro Focus Enterprise Server versione 6.0. Per la versione 8, consulta Configurare Micro Focus Runtime (su Amazon EC2).

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Software e licenza Micro Focus Enterprise Server. Per ulteriori informazioni, contattate il reparto vendite di Micro Focus.
- Comprensione del concetto di ricostruzione e fornitura di un'applicazione mainframe da eseguire in Micro Focus Enterprise Server. Per una panoramica di alto livello, consultate la scheda tecnica di Micro Focus Enterprise Server.
- Comprensione dei concetti di Micro Focus Enterprise Server scale-out Performance and Availability Cluster. Per ulteriori informazioni, consultate la documentazione di Micro Focus Enterprise Server.

- Comprensione del concetto generale di applicazione mainframe DevOps con integrazione continua (CI). Per un modello di guida AWS prescrittiva sviluppato da AWS e Micro Focus, vedi Modernizzazione del mainframe: DevOps on with Micro Focus. AWS

> **Note**
>
> Questo modello è stato testato con Micro Focus Enterprise Server versione 6. Per la versione 8, consulta Configurare Micro Focus Runtime (su Amazon EC2).

Limitazioni

- Per un elenco delle piattaforme supportate da Micro Focus Enterprise Server, consultate la scheda tecnica di Micro Focus Enterprise Server.
- Gli script e i test utilizzati in questo modello si basano su Amazon EC2 Windows Server 2019; altre versioni e sistemi operativi di Windows Server non sono stati testati per questo modello.
- Il modello si basa su Micro Focus Enterprise Server 6.0 per Windows; le versioni precedenti o successive non sono state testate nello sviluppo di questo modello.

Versioni del prodotto

- Micro Focus Enterprise Server 6.0
- Windows Server 2019

Architecture

Nell'ambiente mainframe convenzionale, è necessario fornire l'hardware per ospitare le applicazioni e i dati aziendali. Per far fronte e soddisfare i picchi di richieste stagionali, mensili, trimestrali o addirittura inaspettate o senza precedenti, gli utenti mainframe devono scalare orizzontalmente acquistando capacità di storage e di elaborazione aggiuntive. L'aumento del numero di risorse di archiviazione e capacità di elaborazione migliora le prestazioni complessive, ma la scalabilità non è lineare.

Questo non è il caso quando inizi ad adottare un modello di consumo on demand su AWS utilizzando Amazon Auto EC2 Scaling e Micro Focus Enterprise Server. Le sezioni seguenti spiegano in dettaglio come creare un'architettura applicativa mainframe completamente automatizzata e scalabile

utilizzando Micro Focus Enterprise Server Scale-Out Performance and Availability Cluster (PAC) con un gruppo Amazon Auto Scaling. EC2

Architettura di scalabilità automatica Micro Focus Enterprise Server

Innanzitutto, è importante comprendere i concetti di base di Micro Focus Enterprise Server. Questo ambiente fornisce un ambiente di distribuzione x86 compatibile con il mainframe per le applicazioni tradizionalmente eseguite sul mainframe IBM. Fornisce esecuzioni online e in batch e un ambiente di transazione che supporta quanto segue:

- IBM COBOL
- IBM PL/I
- lavori batch IBM JCL
- Transazioni IBM CICS e IMS TM
- Servizi Web
- Utilità batch comuni, tra cui SORT

Micro Focus Enterprise Server consente l'esecuzione delle applicazioni mainframe con modifiche minime. I carichi di lavoro mainframe esistenti possono essere spostati su piattaforme x86 e modernizzati per sfruttare le estensioni native di AWS Cloud per una rapida espansione verso nuovi mercati o aree geografiche.

Modernizzazione del modello AWS Prescriptive Guidance: su DevOps AWS con Micro Focus ha introdotto l'architettura per accelerare lo sviluppo e il test di applicazioni mainframe su AWS utilizzando Micro Focus Enterprise Developer ed Enterprise Test Server con AWS e AWS. CodePipeline CodeBuild Questo modello si concentra sulla distribuzione di applicazioni mainframe nell'ambiente di produzione AWS per ottenere disponibilità e resilienza elevate.

In un ambiente di produzione mainframe, potresti aver configurato IBM Parallel Sysplex nel mainframe per ottenere prestazioni e disponibilità elevate. Per creare un'architettura scalabile simile a Sysplex, Micro Focus ha introdotto il Performance and Availability Cluster (PAC) su Enterprise Server. PACs supporta la distribuzione di applicazioni mainframe su più regioni Enterprise Server gestite come un'unica immagine e scalabili in istanze Amazon EC2 . PACs supportano anche prestazioni prevedibili delle applicazioni e velocità di trasmissione del sistema su richiesta.

In un PAC, più istanze di Enterprise Server funzionano insieme come un'unica entità logica. Il guasto di un'istanza di Enterprise Server, pertanto, non interromperà la continuità aziendale poiché la capacità è condivisa con altre regioni, mentre le nuove istanze vengono avviate automaticamente

utilizzando funzionalità standard del settore come un gruppo Amazon Auto EC2 Scaling. Ciò rimuove i singoli punti di errore, migliorando la resilienza ai problemi hardware, di rete e delle applicazioni. Le istanze scalabili di Enterprise Server possono essere gestite e gestite utilizzando Enterprise Server Common Web Administration (ESCWA) APIs, che semplifica la manutenzione operativa e la facilità di manutenzione degli Enterprise Server.

> **Note**
>
> Micro Focus consiglia che il Performance and Availability Cluster (PAC) sia composto da almeno tre aree Enterprise Server in modo da non compromettere la disponibilità nel caso in cui un'area Enterprise Server si guasti o richieda manutenzione.

La configurazione PAC richiede un servizio di gestione di database relazionali (RDBMS) supportato per gestire il database regionale, un database interregionale e database di archivi dati opzionali. È necessario utilizzare un database di archivio dati per gestire i file VSAM (Virtual Storage Access Method) utilizzando il supporto Micro Focus Database File Handler per migliorare la disponibilità e la scalabilità. Sono supportati i RDBMSs seguenti:

- Microsoft SQL Server 2009 R2 e versioni successive
- PostgreSQL 10.x, inclusa l'edizione compatibile con Amazon Aurora PostgreSQL
- DB2 10.4 e versioni successive

Per i dettagli sui requisiti RDBMS e PAC supportati, vedere Micro Focus Enterprise Server - Prerequisiti e Micro Focus Enterprise Server - Configurazione PAC consigliata.

Il diagramma seguente mostra una configurazione tipica dell'architettura AWS per un Micro Focus PAC.

	Componente	Descrizione
1	Gruppo di scalabilità automatica delle istanze di Enterprise Server	Configura un gruppo di scalabilità automatico distribuito con istanze di Enterprise Server in un PAC. Il numero

		di istanze può essere scalato orizzontalmente o avviato dagli CloudWatch allarmi di Amazon utilizzando i parametri . CloudWatch
2	Gruppo di scalabilità automatica delle istanze ESCWA di Enterprise Server	Configura un gruppo di ridimensionamento automatico distribuito con Enterprise Server Common Web Administration (ESCWA). ESCWA fornisce la gestione dei cluster. APIs I server ESCWA fungono da piano di controllo per aggiungere o rimuovere Enterprise Server e avviare o arrestare le aree Enterprise Server nel PAC durante gli eventi di scalabilità automatica delle istanze di Enterprise Server. Poiché l'istanza ESCWA viene utilizzata solo per la gestione PAC, il suo schema di traffico è prevedibile e il requisito di capacità desiderato per la scalabilità automatica può essere impostato su 1.
3	Istanza Amazon Aurora in una configurazione Multi-AZ	Configura un sistema di gestione di database relazionali (RDBMS) per ospitare file di dati utente e di sistema da condividere tra le istanze di Enterprise Server.

4	Istanza e replica Amazon ElastiCache (Redis OSS)	Configura un'istanza principale ElastiCache (Redis OSS) e almeno una replica per ospitare i dati degli utenti e fungere da archivio scalabile (SOR) per le istanze di Enterprise Server. È possibile configurare uno o più repository a scalabilità orizzontale per archiviare tipi specifici di dati utente. Enterprise Server utilizza un database Redis NoSQL come SOR, un requisito per mantenere l'integrità PAC.
5	Network Load Balancer	Configura un sistema di bilanciamento del carico, fornendo un nome host alle applicazioni per la connessione ai servizi forniti dalle istanze di Enterprise Server (ad esempio, l'accesso all'applicazione tramite un emulatore 3270).

Questi componenti costituiscono il requisito minimo per un cluster PAC Micro Focus Enterprise Server. La sezione successiva tratta l'automazione della gestione dei cluster.

Utilizzo di AWS Systems Manager Automation per la scalabilità

Dopo la distribuzione del cluster PAC su AWS, il PAC viene gestito tramite Enterprise Server Common Web Administration (ESCWA). APIs

Per automatizzare le attività di gestione dei cluster durante gli eventi di scalabilità automatica, puoi utilizzare i runbook di Systems Manager Automation e Amazon Auto EC2 Scaling con Amazon. EventBridge L'architettura di queste automazioni è mostrata nel diagramma seguente.

	Componente	Descrizione
1	Gancio automatico del ciclo di vita della scalabilità	Configura gli hook automatici del ciclo di vita della scalabilità e invia notifiche ad Amazon EventBridge quando vengono lanciate nuove istanze e le istanze esistenti vengono terminate nel gruppo di scalabilità automatica.
2	Amazon EventBridge	Imposta una EventBridge regola Amazon per indirizzare gli eventi di scalabilità automatica verso le destinazioni del runbook di Systems Manager Automation.
3	Runbook di automazione	Configura i runbook di Systems Manager Automation per eseguire PowerShell gli script di Windows e richiama ESCWA APIs per gestire il PAC. Per alcuni esempi, vedere la sezione Informazioni aggiuntive.
4	Istanza ESCWA di Enterprise Server in un gruppo di scalabilità automatica	Configurare un'istanza ESCWA di Enterprise Server in un gruppo di ridimensionamento automatico. L'istanza ESCWA consente di APIs gestire il PAC.

Tools (Strumenti)

- Micro Focus Enterprise Server — Micro Focus Enterprise Server fornisce l'ambiente di esecuzione per le applicazioni create con qualsiasi variante dell'ambiente di sviluppo integrato (IDE) di Enterprise Developer.
- Amazon EC2 Auto Scaling: Amazon EC2 Auto Scaling ti aiuta a garantire il numero corretto di istanze EC2 Amazon disponibili per gestire il carico della tua applicazione. È possibile creare raccolte di EC2 istanze, denominate gruppi di Auto Scaling, e specificare il numero minimo e massimo di istanze.
- Amazon ElastiCache (Redis OSS): Amazon ElastiCache è un servizio web per configurare, gestire e scalare un archivio dati distribuito in memoria o un ambiente di cache nel cloud. Fornisce una soluzione di caching scalabile ad alte prestazioni e a costi contenuti.
- Amazon RDS — Amazon Relational Database Service (Amazon RDS) è un servizio Web che semplifica la configurazione, il funzionamento e la scalabilità di un database relazionale nel cloud AWS. Fornisce una capacità ridimensionabile e conveniente per un database relazionale e gestisce le attività di amministrazione comuni del database.
- AWS Systems Manager — AWS Systems Manager è un servizio AWS che puoi usare per visualizzare e controllare la tua infrastruttura su AWS. Utilizzando la console Systems Manager, puoi visualizzare i dati operativi da più servizi AWS e automatizzare le attività operative tra le tue risorse AWS. Systems Manager consente di mantenere la sicurezza e la conformità eseguendo la scansione delle Istanze gestite e segnalando eventuali violazioni dei criteri rilevate (o intraprendendo azioni correttive in merito).

Epiche

Crea un'istanza Amazon Aurora

Operazione	Description	Competenze richieste
Crea un CloudFormation modello AWS per un'istanza Amazon Aurora.	Utilizza lo snippet di codice di esempio di AWS per creare un CloudFormation modello che creerà un'istanza Edition compatibile con Amazon Aurora PostgreSQL.	Architetto del cloud

Operazione	Description	Competenze richieste
Implementa uno CloudForm ation stack per creare l'istanza Amazon Aurora.	Usa il CloudFormation modello per creare un'istanz a compatibile con Aurora PostgreSQL con la replica Multi-AZ abilitata per i carichi di lavoro di produzione.	Architetto del cloud
Configurare le impostazioni di connessione al database per Enterprise Server.	Seguite le istruzioni nella documentazione di Micro Focus per preparare le stringhe di connessione e la configurazione del database per Micro Focus Enterprise Server.	Ingegnere dei dati, DevOps ingegnere

Crea un ElastiCache cluster Amazon per l'istanza Redis

Operazione	Description	Competenze richieste
Crea un CloudFormation modello per il ElastiCache cluster Amazon per l'istanza Redis.	Usa lo snippet di codice di esempio di AWS per creare un CloudFormation modello che creerà un ElastiCache cluster Amazon per l'istanza Redis.	Architetto del cloud
Implementa lo CloudForm ation stack per creare un ElastiCache cluster Amazon per l'istanza Redis.	Crea il ElastiCache cluster Amazon per l'istanza Redis con la replica Multi-AZ abilitata per i carichi di lavoro di produzione.	Architetto del cloud
Configurare le impostazioni di connessione PSOR di Enterprise Server.	Seguite le istruzioni nella documentazione di Micro Focus per preparare la configurazione di connessio	DevOps ingegnere

Operazione	Description	Competenze richieste
	ne PAC Scale-Out Repository (PSOR) per Micro Focus Enterprise Server PAC.	

Create un gruppo di scalabilità automatico Micro Focus Enterprise Server ESCWA

Operazione	Description	Competenze richieste
Create un'AMI Micro Focus Enterprise Server.	Crea un'istanza Amazon EC2 Windows Server e installa il file binario Micro Focus Enterprise Server nell' EC2 istanza. Crea un'Amazon Machine Image (AMI) dell' EC2 istanza. Per ulteriori informazioni, consulta la documentazione di installazione di Enterprise Server.	Architetto del cloud
Crea un CloudFormation modello per Enterprise Server ESCWA.	Usa lo snippet di codice di esempio di AWS per creare un modello per creare uno stack personalizzato di Enterprise Server ESCWA in un gruppo di ridimensionamento automatico.	Architetto del cloud
Implementa lo CloudFormation stack per creare un gruppo di EC2 scalabilità Amazon per Enterprise Server ESCWA.	Utilizzate il CloudFormation modello per implementare il gruppo di scalabilità automatico con l'AMI ESCWA Micro Focus Enterprise Server creato nella storia precedente.	Architetto del cloud

Crea un runbook di AWS Systems Manager Automation

Operazione	Description	Competenze richieste
Crea un CloudFormation modello per un runbook di Systems Manager Automation.	Utilizza i frammenti di codice di esempio nella sezione Informazioni aggiuntive per creare un CloudForm ation modello che creerà un runbook di Systems Manager Automation per automatiz zare la creazione di PAC, la scalabilità iniziale di Enterpris e Server e la scalabilità orizzontale di Enterprise Server.	Architetto del cloud
Implementa lo CloudFormation stack che contiene il runbook Systems Manager Automation.	Utilizzate il CloudFormation modello per distribuire uno stack che contenga il runbook Automation per la creazione di PAC, Enterprise Server scalabilità in ed Enterprise Server scalabilità out.	Architetto del cloud

Create un gruppo di scalabilità automatico per Micro Focus Enterprise Server

Operazione	Description	Competenze richieste
Create un CloudFormation modello per configurare un gruppo di scalabilità automatic o per Micro Focus Enterprise Server.	Usa lo snippet di codice di esempio di AWS per creare un CloudFormation modello che creerà un gruppo di scalabili tà automatico. Questo modello riutilizzerà la stessa AMI creata per l'istanza ESCWA	Architetto del cloud

Operazione	Description	Competenze richieste
	di Micro Focus Enterprise Server. Utilizza quindi un frammento di codice di esempio di AWS per creare l'evento di scalabilità automatica del ciclo di vita e configura Amazon per EventBridge filtrare gli eventi di scalabilità orizzontale e orizzontale nello stesso modello. CloudFormation	
Implementate lo CloudFormation stack per il gruppo di scalabilità automatico per i server Micro Focus Enterprise.	Implementate lo CloudFormation stack che contiene il gruppo di scalabilità automatico per Micro Focus Enterprise Server.	Architetto del cloud

Risorse correlate

- Cluster di prestazioni e disponibilità dei server Micro Focus Enterprise (PAC)
- Ganci per il ciclo di vita di Amazon EC2 Auto Scaling
- Esecuzione di automazioni con trigger utilizzando EventBridge

Informazioni aggiuntive

I seguenti scenari devono essere automatizzati per la scalabilità interna o orizzontale dei cluster PAC.

Automazione per l'avvio o la ricreazione di un PAC

All'avvio di un cluster PAC, Enterprise Server richiede che ESCWA venga richiamato per APIs creare una configurazione PAC. Questo avvia e aggiunge le regioni Enterprise Server al PAC. Per creare o ricreare un PAC, attenersi alla seguente procedura:

1. Configura un PAC Scale-Out Repository (PSOR) in ESCWA con un determinato nome.

```
POST /server/v1/config/groups/sors
```

2. Crea un PAC con un determinato nome e allega il PSOR ad esso.

```
POST /server/v1/config/groups/pacs
```

3. Configura il database regionale e il database interregionale se è la prima volta che configuri un PAC.

> **Note**
>
> Questo passaggio utilizza le query SQL e lo strumento dbhfhadmin da riga di comando di Micro Focus Enterprise Suite per creare il database e importare i dati iniziali.

4. Installa la definizione PAC nelle regioni Enterprise Server.

```
POST /server/v1/config/mfds
POST /native/v1/config/groups/pacs/${pac_uid}/install
```

5. Avvia le regioni Enterprise Server nel PAC.

```
POST /native/v1/regions/${host_ip}/${port}/${region_name}/start
```

I passaggi precedenti possono essere implementati utilizzando uno PowerShell script di Windows.

I passaggi seguenti spiegano come creare un'automazione per la creazione di un PAC riutilizzando lo script di Windows PowerShell .

1. Crea un modello di EC2 avvio Amazon che scarichi o crei PowerShell lo script di Windows come parte del processo di bootstrap. Ad esempio, puoi utilizzare i dati EC2 utente per scaricare lo script da un bucket Amazon Simple Storage Service (Amazon S3).
2. Crea un runbook AWS Systems Manager Automation per richiamare lo script di Windows PowerShell .
3. Associa il runbook all'istanza ESCWA utilizzando il tag di istanza.
4. Crea un gruppo di ridimensionamento automatico ESCWA utilizzando il modello di avvio.

Puoi utilizzare lo CloudFormation snippet AWS di esempio seguente per creare il runbook di automazione.

CloudFormation Frammento di esempio per un runbook di Systems Manager Automation utilizzato per la creazione di PAC

```
PACInitDocument:
   Type: AWS::SSM::Document
   Properties:
     DocumentType: Command
     Content:
       schemaVersion: '2.2'
       description: Operation Runbook to create Enterprise Server PAC
       mainSteps:
       - action: aws:runPowerShellScript
         name: CreatePAC
         inputs:
           onFailure: Abort
           timeoutSeconds: "1200"
           runCommand:
           - |
             C:\Scripts\PAC-Init.ps1
PacInitAutomation:
   Type: AWS::SSM::Document
   Properties:
     DocumentType: Automation
     Content:
       description: Prepare Micro Focus PAC Cluster via ESCWA Server
       schemaVersion: '0.3'
       assumeRole: !GetAtt SsmAssumeRole.Arn
       mainSteps:
         - name: RunPACInitDocument
           action: aws:runCommand
           timeoutSeconds: 300
           onFailure: Abort
           inputs:
             DocumentName: !Ref PACInitDocument
             Targets:
               - Key: tag:Enterprise Server - ESCWA
                 Values:
                   - "true"
 PacInitDocumentAssociation:
   Type: AWS::SSM::Association
```

```
    Properties:
      DocumentVersion: "$LATEST"
      Name: !Ref PACInitDocument
      Targets:
        - Key: tag:Enterprise Server - ESCWA
          Values:
            - "true"
```

Per ulteriori informazioni, vedere Micro Focus Enterprise Server - Configurazione di un PAC.

Automazione per la scalabilità orizzontale con una nuova istanza di Enterprise Server

Quando un'istanza Enterprise Server viene scalata orizzontalmente, la relativa regione Enterprise Server deve essere aggiunta al PAC. I passaggi seguenti spiegano come richiamare ESCWA APIs e aggiungere la regione Enterprise Server al PAC.

1. Installare la definizione PAC nelle regioni Enterprise Server.

```
POST '/server/v1/config/mfds'
POST /native/v1/config/groups/pacs/${pac_uid}/install
```

2. Warm Start la regione nel PAC.

```
POST /native/v1/regions/${host_ip}/${port}/${region_name}/start
```

3. Aggiungere l'istanza Enterprise Server al load balancer associando il gruppo di scalabilità automatica al load balancer.

I passaggi precedenti possono essere implementati utilizzando uno script di Windows. PowerShell Per ulteriori informazioni, vedere Micro Focus Enterprise Server - Configurazione di un PAC.

I seguenti passaggi possono essere utilizzati per creare un'automazione basata sugli eventi per aggiungere un'istanza Enterprise Server appena lanciata in un PAC riutilizzando lo script di Windows. PowerShell

1. Crea un modello di EC2 avvio Amazon per un'istanza di Enterprise Server che effettui il provisioning di una regione Enterprise Server durante il suo avvio. Ad esempio, è possibile utilizzare il comando Micro Focus Enterprise Server mfds per importare una configurazione regionale. Per ulteriori dettagli e opzioni disponibili per questo comando, consultate Enterprise Server Reference.

2. Creare un gruppo di scalabilità automatica di Enterprise Server che utilizzi il modello di avvio creato nel passaggio precedente.
3. Crea un runbook di Systems Manager Automation per richiamare lo script di Windows PowerShell .
4. Associate il runbook all'istanza ESCWA utilizzando il tag instance.
5. Crea una EventBridge regola Amazon per filtrare l'evento EC2 Instance Launch Successful per il gruppo di scalabilità automatica Enterprise Server e crea la destinazione per utilizzare il runbook di automazione.

È possibile utilizzare lo CloudFormation snippet di esempio seguente per creare il runbook di automazione e la regola. EventBridge

CloudFormation Frammento di esempio per Systems Manager utilizzato per scalare le istanze di Enterprise Server

```
ScaleOutDocument:
   Type: AWS::SSM::Document
   Properties:
     DocumentType: Command
     Content:
       schemaVersion: '2.2'
       description: Operation Runbook to Adding MFDS Server into an existing PAC
       parameters:
         MfdsPort:
           type: String
         InstanceIpAddress:
           type: String
           default: "Not-Available"
         InstanceId:
           type: String
           default: "Not-Available"
       mainSteps:
       - action: aws:runPowerShellScript
         name: Add_MFDS
         inputs:
           onFailure: Abort
           timeoutSeconds: "300"
           runCommand:
           - |
             $ip = "{{InstanceIpAddress}}"
```

```
              if ( ${ip} -eq "Not-Available" ) {
                $ip = aws ec2 describe-instances --instance-id {{InstanceId}} --output
text --query "Reservations[0].Instances[0].PrivateIpAddress"
              }
              C:\Scripts\Scale-Out.ps1 -host_ip ${ip} -port {{MfdsPort}}

  PacScaleOutAutomation:
    Type: AWS::SSM::Document
    Properties:
      DocumentType: Automation
      Content:
        parameters:
          MfdsPort:
            type: String
          InstanceIpAddress:
            type: String
            default: "Not-Available"
          InstanceId:
            type: String
            default: "Not-Available"
        description: Scale Out 1 New Server in Micro Focus PAC Cluster via ESCWA
Server
        schemaVersion: '0.3'
        assumeRole: !GetAtt SsmAssumeRole.Arn
        mainSteps:
          - name: RunScaleOutCommand
            action: aws:runCommand
            timeoutSeconds: 300
            onFailure: Abort
            inputs:
              DocumentName: !Ref ScaleOutDocument
              Parameters:
                InstanceIpAddress: "{{InstanceIpAddress}}"
                InstanceId: "{{InstanceId}}"
                MfdsPort: "{{MfdsPort}}"
              Targets:
                - Key: tag:Enterprise Server - ESCWA
                  Values:
                    - "true"
```

Automazione per la scalabilità in un'istanza di Enterprise Server

Analogamente alla scalabilità orizzontale, quando un'istanza di Enterprise Server viene scalata in orizzontale, viene avviato l'evento EC2 Instance-terminate Lifecycle Action e sono necessarie le seguenti chiamate di processo e API per rimuovere un'istanza di Micro Focus Enterprise Server dal PAC.

1. Arresta la regione nell'istanza di Enterprise Server in fase di terminazione.

```
POST "/native/v1/regions/${host_ip}/${port}/${region_name}/stop"
```

2. Rimuovere l'istanza Enterprise Server dal PAC.

```
DELETE "/server/v1/config/mfds/${uid}"
```

3. Invia un segnale per continuare a terminare l'istanza di Enterprise Server.

I passaggi precedenti possono essere implementati in uno PowerShell script di Windows. Per ulteriori dettagli su questo processo, consultate il documento Micro Focus Enterprise Server - Amministrazione di un PAC.

I passaggi seguenti spiegano come creare un'automazione basata sugli eventi per terminare un'istanza di Enterprise Server da un PAC riutilizzando lo script di Windows. PowerShell

1. Crea un runbook di Systems Manager Automation per richiamare lo script di Windows PowerShell .
2. Associate il runbook all'istanza ESCWA utilizzando il tag instance.
3. Crea un hook automatico per il ciclo di vita del gruppo con scalabilità, ad esempio la terminazione. EC2
4. Crea una EventBridge regola Amazon per filtrare l'evento EC2 Instance-terminate Lifecycle Action per il gruppo di scalabilità automatica di Enterprise Server e crea la destinazione per utilizzare il runbook di automazione.

È possibile utilizzare il seguente CloudFormation modello di esempio per creare un runbook, un hook del ciclo di vita e una regola di Systems Manager Automation. EventBridge

CloudFormation Frammento di esempio per un runbook di Systems Manager Automation utilizzato per la scalabilità in un'istanza di Enterprise Server

```
  ScaleInDocument:
```

```
    Type: AWS::SSM::Document
    Properties:
      DocumentType: Command
      Content:
        schemaVersion: '2.2'
        description: Operation Runbook to Remove MFDS Server from PAC
        parameters:
          MfdsPort:
            type: String
          InstanceIpAddress:
            type: String
            default: "Not-Available"
          InstanceId:
            type: String
            default: "Not-Available"
        mainSteps:
        - action: aws:runPowerShellScript
          name: Remove_MFDS
          inputs:
            onFailure: Abort
            runCommand:
            - |
              $ip = "{{InstanceIpAddress}}"
              if ( ${ip} -eq "Not-Available" ) {
                $ip = aws ec2 describe-instances --instance-id {{InstanceId}} --output
text --query "Reservations[0].Instances[0].PrivateIpAddress"
              }
              C:\Scripts\Scale-In.ps1 -host_ip ${ip} -port {{MfdsPort}}

  PacScaleInAutomation:
    Type: AWS::SSM::Document
    Properties:
      DocumentType: Automation
      Content:
        parameters:
          MfdsPort:
            type: String
          InstanceIpAddress:
            type: String
            default: "Not-Available"
          InstanceId:
            type: String
            default: "Not-Available"
        description: Scale In 1 New Server in Micro Focus PAC Cluster via ESCWA Server
```

```
          schemaVersion: '0.3'
          assumeRole: !GetAtt SsmAssumeRole.Arn
          mainSteps:
            - name: RunScaleInCommand
              action: aws:runCommand
              timeoutSeconds: "600"
              onFailure: Abort
              inputs:
                DocumentName: !Ref ScaleInDocument
                Parameters:
                  InstanceIpAddress: "{{InstanceIpAddress}}"
                  MfdsPort: "{{MfdsPort}}"
                  InstanceId: "{{InstanceId}}"
                Targets:
                  - Key: tag:Enterprise Server - ESCWA
                    Values:
                      - "true"
            - name: TerminateTheInstance
              action: aws:executeAwsApi
              inputs:
                Service: autoscaling
                Api: CompleteLifecycleAction
                AutoScalingGroupName: !Ref AutoScalingGroup
                InstanceId: "{{ InstanceId }}"
                LifecycleActionResult: CONTINUE
                LifecycleHookName: !Ref ScaleInLifeCycleHook
```

Automazione per un trigger di scalabilità EC2 automatica di Amazon

Il processo di impostazione di una politica di scalabilità per le istanze di Enterprise Server richiede una comprensione del comportamento dell'applicazione. Nella maggior parte dei casi, è possibile impostare politiche di scalabilità di Target Tracking. Ad esempio, puoi utilizzare l'utilizzo medio della CPU come CloudWatch metrica Amazon per impostare la politica di scalabilità automatica. Per ulteriori informazioni, consulta le politiche di scalabilità di monitoraggio di Target per Amazon EC2 Auto Scaling. Per le applicazioni con schemi di traffico regolari, prendi in considerazione l'utilizzo di una politica di scalabilità predittiva. Per ulteriori informazioni, consulta Scaling predittivo per Amazon Auto EC2 Scaling.

Crea un visualizzatore di file mainframe avanzato nel cloud AWS

Boopathy GOPALSAMY e Jeremiah O'Connor, Amazon Web Services

Riepilogo

Questo modello fornisce esempi di codice e passaggi per aiutarti a creare uno strumento avanzato per la navigazione e la revisione dei file mainframe in formato fisso utilizzando i servizi serverless AWS. Il modello fornisce un esempio di come convertire un file di input mainframe in un documento Amazon OpenSearch Service per la navigazione e la ricerca. Lo strumento di visualizzazione dei file può aiutarti a ottenere quanto segue:

- Mantieni la stessa struttura e lo stesso layout dei file mainframe per garantire la coerenza nell'ambiente di migrazione di destinazione AWS (ad esempio, puoi mantenere lo stesso layout per i file in un'applicazione batch che trasmette i file a parti esterne)
- Velocizza lo sviluppo e i test durante la migrazione del mainframe
- Supporta le attività di manutenzione dopo la migrazione

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un cloud privato virtuale (VPC) con una sottorete raggiungibile dalla tua piattaforma legacy
-

> **Note**
>
> Un file di input e il corrispondente quaderno COBOL (Common Business-Oriented Language) (: Per esempi di file di input e quaderni COBOL, consultate il repository. gfs-mainframe-solutions GitHub Per ulteriori informazioni sui quaderni COBOL, consulta la Guida alla programmazione di Enterprise COBOL for 6.3 sul sito Web IBM.) z/OS

Limitazioni

- L'analisi dei copybook è limitata a non più di due livelli annidati (OCCURS)

Architecture

Stack di tecnologia di origine

- File di input in formato FB (Fixed Blocked)
- Layout del quaderno COBOL

Stack tecnologico Target

- Amazon Athena
- OpenSearch Servizio Amazon
- Amazon Simple Storage Service (Amazon S3)
- AWS Lambda
- AWS Step Functions

Architettura Target

Il diagramma seguente mostra il processo di analisi e conversione di un file di input del mainframe in un documento di OpenSearch servizio per la navigazione e la ricerca.

Il diagramma mostra il flusso di lavoro seguente:

1. Un utente o un'applicazione amministratore invia i file di input a un bucket S3 e i quaderni COBOL a un altro bucket S3.
2.
> **Note**
>
> Il bucket S3 con i file di input richiama una funzione Lambda che avvia un flusso di lavoro Step Functions senza server. : L'uso di un trigger di eventi S3 e della funzione Lambda per guidare il flusso di lavoro Step Functions in questo modello è facoltativo. Gli esempi di GitHub codice di questo modello non includono l'uso di questi servizi, ma è possibile utilizzarli in base alle proprie esigenze.

3. Il flusso di lavoro Step Functions coordina tutti i processi batch delle seguenti funzioni Lambda:
   - La `s3copybookparser.py` funzione analizza il layout del quaderno ed estrae gli attributi dei campi, i tipi di dati e gli offset (necessari per l'elaborazione dei dati di input).

- La `s3toathena.py` funzione crea un layout di tabella Athena. Athena analizza i dati di input elaborati dalla `s3toathena.py` funzione e li converte in un file CSV.
- La `s3toelasticsearch.py` funzione acquisisce il file dei risultati dal bucket S3 e lo invia al servizio. OpenSearch

4. Gli utenti accedono a OpenSearch Dashboards with OpenSearch Service per recuperare i dati in vari formati di tabelle e colonne e quindi eseguire query sui dati indicizzati.

Tools (Strumenti)

Servizi AWS

- Amazon Athena è un servizio di query interattivo che ti aiuta ad analizzare i dati direttamente in Amazon Simple Storage Service (Amazon S3) utilizzando SQL standard.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi. In questo modello, si utilizza Lambda per implementare la logica di base, come l'analisi dei file, la conversione dei dati e il caricamento dei dati in OpenSearch Service per l'accesso interattivo ai file.
- Amazon OpenSearch Service è un servizio gestito che ti aiuta a distribuire, gestire e scalare i cluster OpenSearch di servizi nel cloud AWS. In questo modello, utilizzi OpenSearch Service per indicizzare i file convertiti e fornire funzionalità di ricerca interattive per gli utenti.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Step Functions è un servizio di orchestrazione senza server che ti aiuta a combinare funzioni Lambda e altri servizi AWS per creare applicazioni aziendali critiche. In questo modello, usi Step Functions per orchestrare le funzioni Lambda.

Altri strumenti

- GitHubè un servizio di code-hosting che fornisce strumenti di collaborazione e controllo delle versioni.

- Python è un linguaggio di programmazione di alto livello.

Codice

Il codice per questo pattern è disponibile nel repository. GitHub gfs-mainframe-patterns

Epiche

Prepara l'ambiente di destinazione

Operazione	Description	Competenze richieste
Crea il bucket S3.	Crea un bucket S3 per archiviare i quaderni, i file di input e i file di output. Ti consigliamo la seguente struttura di cartelle per il tuo bucket S3: • `copybook/` • `input/` • `output/` • `query/` • `results/`	Informazioni generali su AWS
Crea la funzione s3copyboo kparser.	1. Crea una funzione Lambda chiamata `s3copyboo kparser` e carica il codice sorgente (`s3copybookparser.p y` `ecopybook.py`) dal GitHubrepository. 2. Associa la policy IAM `S3ReadOnly` alla funzione Lambda.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Crea la funzione s3toathena.	1. Crea una funzione Lambda chiamata `s3toathena` e carica il codice sorgente (`s3toathena.py`) dal GitHubrepository. Configura il timeout Lambda su > 60 secondi. 2. Per fornire l'accesso alle risorse richieste, collega le policy IAM `AmazonAthenaFullAccess` e `S3FullAccess` la funzione Lambda.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Crea la funzione s3toelasticsearch.	1. **Important** Aggiungi una dipendenza Python al tuo ambiente Lambda. : per utilizzare la `s3toelasticsearch` funzione, è necessario aggiungere la dipendenza Python perché la funzione Lambda utilizza le dipendenze del client Python Elasticsearch (e). `Elasticsearch==7.9.0 requests_aws4auth` 2. Crea una funzione Lambda chiamata `s3toelasticsearch` e carica il codice sorgente (`s3toelasticsearch.py`) dal GitHubrepository. 3. Importa la dipendenz a Python come livello Lambda. 4. Allega le policy IAM `S3ReadOnly` e	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	`AmazonOpenSearchServiceReadOnlyAccess` alla funzione Lambda.	

Operazione	Description	Competenze richieste
Crea il cluster OpenSearch di servizi.	Crea il cluster 1. Crea un cluster OpenSearch di servizi. Quando crei il cluster, procedi come segue: • **Note** Crea un utente principale e una password per il cluster da utilizzare per accedere alle OpenSearch dashboard. : Questo passaggio non è necessario se utilizzi l'autenticazione tramite Amazon Cognito. • Scegli un controllo granulare degli accessi. Questo ti offre ulteriori modi per controllare l'accesso ai tuoi dati in Service. OpenSearch 2. Copia l'URL del dominio e passalo come variabile di ambiente 'HOST' alla funzione Lambda. `s3toelasticsearch`	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	Concedi l'accesso al ruolo IAM Per fornire un accesso granulare al ruolo IAM (`arn:aws:iam::**:role/service-role/s3toelasticsearch-role-**`) della funzione Lambda, procedi come segue: 1. Accedi a OpenSearch Dashboards come utente principale. 2. Scegli la scheda Sicurezza , quindi scegli Roles, all_access, Map user, Backend roles. 3. Aggiungi l'Amazon Resource Name (ARN) del ruolo IAM della funzione Lambda, quindi scegli Salva. Per ulteriori informazioni, consulta Mappatura dei ruoli agli utenti nella documentazione del OpenSearch servizio.	

Operazione	Description	Competenze richieste
Crea Step Functions per l'orchestrazione.	1. Crea una macchina a stati Step Functions con il flusso standard. La definizione è inclusa nel GitHub repository. 2. Nello script JSON, sostituisci la funzione Lambda con ARNs ARNs la funzione from Lambda del tuo ambiente.	Informazioni generali su AWS

Implementa ed esegui

Operazione	Description	Competenze richieste
Carica i file di input e i quaderni nel bucket S3.	Scarica i file di esempio dalla cartella di esempio del GitHub repository e carica i file nel bucket S3 che hai creato in precedenza. 1. Carica `Mockedcopy.cpy` e nella cartella`acctix.cpy .<S3_Bucket>/copybook` 2. Carica i file `acctindex.cpy` di input `Modedupdate.txt` e gli esempi nella `<S3_Bucket>/input` cartella.	Informazioni generali su AWS
Invoca Step Functions.	1. Accedi alla Console di gestione AWS e apri la console Step Functions.	Informazioni generali su AWS

Operazione	Description	Competenze richieste

	2. Nel pannello di navigazione, scegli Macchine a stati.
	3. Scegli la tua macchina a stati, quindi scegli Avvia esecuzione.
	4. Nella casella Input, inserisci il seguente copybook/file percorso come variabile JSON per il bucket S3, quindi scegli Avvia esecuzione.
	
	`{ "s3_copybook_bucket_name": "<BUCKET NAME>", "s3_copybook_bucket_key": "<COPYBOOK PATH>", "s3_source_bucket_name": "<BUCKET NAME", "s3_source_bucket_key": "INPUT FILE PATH" }`
	
	Esempio:
	
	`{ "s3_copybook_bucket_name": "fileaidtest", "s3_copybook_bucket_key": "copybook/acctix.cpy", "s3_source_bucket_name": "fileaidtest",`	

Operazione	Description	Competenze richieste

	<pre>"s3_source_bucket_key": "input/actindex"
	}</pre>	

Convalida l'esecuzione del flusso di lavoro in Step Functions.	Nella console Step Functions, esaminate l'esecuzione del flusso di lavoro nell'ispettore Graph. Gli stati di esecuzione sono codificati a colori per rappresentare lo stato di esecuzione. Ad esempio, il blu indica In corso, il verde indica Riuscito e il rosso indica Non riuscito. È inoltre possibile consultare la tabella nella sezione Cronologia degli eventi di esecuzione per informazioni più dettagliate sugli eventi di esecuzione. Per un esempio di esecuzione grafica del flusso di lavoro, vedete il grafico Step Functions nella sezione Informazioni aggiuntive di questo modello.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Convalida i registri di spedizione in Amazon. CloudWatch	1. Accedi alla console di gestione AWS e apri la console CloudWatch . 2. Nel riquadro di navigazione, espandi Registri, quindi scegli Gruppi di log. 3. Nella casella di ricerca, cerca il gruppo di log della `s3toelasticsearch` funzione. Per un esempio di registri di consegna riusciti, consulta i registri di CloudWatch consegna nella sezione Informazioni aggiuntive di questo modello.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Convalida il file formattato nelle OpenSearch dashboard ed esegui operazioni sui file.	1. Accedi alla Console di gestione AWS. In Analytics , scegli Amazon OpenSearch Service. 2. Nel riquadro di navigazione, scegli Domini. 3. Nella casella di ricerca, inserisci l'URL del tuo dominio in OpenSearch Dashboards. 4. Scegli la tua dashboard, quindi accedi come utente principale. 5. Sfoglia i dati indicizzati in formato tabella. 6. Confronta il file di input con il file di output formattato (documento indicizzato) nelle dashboard. OpenSearch La visualizzazione dashboard mostra le intestazioni di colonna aggiunte per i file formattati. Verifica che i dati di origine dei file di input non formattati corrispondano ai dati di destinazione nella visualizzazione del dashboard. 7. Esegui azioni come la ricerca (ad esempio, utilizzando nomi di campo, valori o espressioni),	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	il filtro e le operazioni DQL (Dashboard Query Language) sul file indicizza to.	

Risorse correlate

Riferimenti

- Esempio di quaderno COBOL (documentazione IBM)
- BMC Compuware File-AID (documentazione BMC)

Tutorial

- Tutorial: utilizzo di un trigger Amazon S3 per richiamare una funzione Lambda (documentazione AWS Lambda)
- Come posso creare un flusso di lavoro serverless con AWS Step Functions e AWS Lambda (documentazione AWS)
- Utilizzo di OpenSearch dashboard con Amazon OpenSearch Service (documentazione AWS)

Informazioni aggiuntive

Grafico Step Functions

L'esempio seguente mostra un grafico Step Functions. Il grafico mostra lo stato di esecuzione delle funzioni Lambda utilizzate in questo modello.

CloudWatch registri di consegna

L'esempio seguente mostra i log di consegna riusciti per l'esecuzione dell'`s3toelasticsearch`esecuzione.

2022-08-10T 15:53:33. 033-05:00	Numero di documenti di elaborazione: 100

2022-08-10T 15:53:33. 171-05:00	[INFORMAZIONI] 2022-08-1 0T 20:53:33.171 Z a1b2c3d4- 5678-90ab-cdef- Post:443/ _bulk [stato:200 richiesta :0.100s] EXAMPLE11111 https://search-essearch-3h4 uqclifeqaj2vg4mphe7ffle.us- east-2.es.amazonaws.com
2022-08-10T 15:53:33. 172-05:00	Scrittura in blocco riuscita: 100 documenti

Containerizza i carichi di lavoro mainframe che sono stati modernizzati da Blu Age

Richard Milner-Watts, Amazon Web Services

Riepilogo

Questo modello fornisce un ambiente container di esempio per l'esecuzione di carichi di lavoro mainframe che sono stati modernizzati utilizzando lo strumento Blu Age. Blu Age converte i carichi di lavoro mainframe legacy in codice Java moderno. Questo modello fornisce un wrapper per l'applicazione Java in modo da poterla eseguire utilizzando servizi di orchestrazione di container come Amazon Elastic Container Service (Amazon ECS) o Amazon Elastic Kubernetes Service (Amazon EKS).

Per ulteriori informazioni sulla modernizzazione dei carichi di lavoro utilizzando Blu Age e i servizi AWS, consulta queste pubblicazioni di AWS Prescriptive Guidance:

- Esecuzione di carichi di lavoro mainframe Blu Age modernizzati su un'infrastruttura AWS serverless
- Implementa un ambiente per applicazioni Blu Age containerizzate utilizzando Terraform

Per assistenza sull'utilizzo di Blu Age per modernizzare i carichi di lavoro mainframe, contatta il team Blu Age scegliendo Contatta i nostri esperti sul sito Web Blu Age. Per assistenza sulla migrazione dei carichi di lavoro modernizzati su AWS, l'integrazione con i servizi AWS e il loro trasferimento in produzione, contatta il tuo account manager AWS o compila il modulo AWS Professional Services.

Prerequisiti e limitazioni

Prerequisiti

- Un'applicazione Java modernizzata creata da Blu Age. A scopo di test, questo modello fornisce un'applicazione Java di esempio che è possibile utilizzare come proof of concept.
- Un ambiente Docker che puoi usare per creare il contenitore.

Limitazioni

A seconda della piattaforma di orchestrazione dei container utilizzata, le risorse che possono essere rese disponibili al contenitore (come CPU, RAM e storage) potrebbero essere limitate. Ad esempio,

se utilizzi Amazon ECS con AWS Fargate, consulta la documentazione di Amazon ECS per limiti e considerazioni.

Architecture

Stack tecnologico di origine

- Età blu
- Java

Stack tecnologico Target

- Docker

Architettura di destinazione

Il diagramma seguente mostra l'architettura dell'applicazione Blu Age all'interno di un contenitore Docker.

1. Il punto di ingresso per il contenitore è lo script wrapper. Questo script bash è responsabile della preparazione dell'ambiente di runtime per l'applicazione Blu Age e dell'elaborazione degli output.
2. Le variabili di ambiente all'interno del contenitore vengono utilizzate per configurare le variabili nello script wrapper, come i nomi dei bucket di Amazon Simple Storage Service (Amazon S3) e le credenziali del database. Le variabili di ambiente sono fornite da AWS Secrets Manager o Parameter Store, una funzionalità di AWS Systems Manager. Se utilizzi Amazon ECS come servizio di orchestrazione dei container, puoi anche codificare le variabili di ambiente nella definizione del task di Amazon ECS.
3. Lo script wrapper è responsabile dell'estrazione di tutti i file di input dal bucket S3 nel contenitore prima di eseguire l'applicazione Blu Age. L'AWS Command Line Interface (AWS CLI) viene installata all'interno del contenitore. Ciò fornisce un meccanismo per accedere agli oggetti archiviati in Amazon S3 tramite l'endpoint gateway Virtual Private Cloud (VPC).
4. Il file Java Archive (JAR) per l'applicazione Blu Age potrebbe dover comunicare con altre fonti di dati, come Amazon Aurora.
5. Dopo il completamento, lo script wrapper invia i file di output risultanti in un bucket S3 per un'ulteriore elaborazione (ad esempio, tramite Amazon CloudWatch Logging Services). Il

modello supporta anche l'invio di file di log compressi ad Amazon S3, se utilizzi un'alternativa alla CloudWatch registrazione standard.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- Amazon Elastic Container Service (Amazon ECS) è un servizio rapido e scalabile di gestione dei container che ti aiuta a eseguire, arrestare e gestire container in un cluster.

Strumenti

- Docker è una piattaforma software per la creazione, il test e la distribuzione di applicazioni. Docker impacchetta il software in unità standardizzate chiamate contenitori, che contengono tutto ciò di cui il software ha bisogno per funzionare, tra cui librerie, strumenti di sistema, codice e runtime. Puoi usare Docker per distribuire e scalare le applicazioni in qualsiasi ambiente.
- Bash è un'interfaccia in linguaggio di comando (shell) per il sistema operativo GNU.
- Java è il linguaggio di programmazione e l'ambiente di sviluppo utilizzati in questo modello.
- Blu Age è uno strumento di modernizzazione dei mainframe AWS che converte i carichi di lavoro mainframe legacy, inclusi codice applicativo, dipendenze e infrastruttura, in carichi di lavoro moderni per il cloud.

Repository di codice

Il codice per questo pattern è disponibile nel repository di contenitori di esempio GitHub Blu Age.

Best practice

- Esternalizza le variabili per modificare il comportamento dell'applicazione utilizzando variabili di ambiente. Queste variabili consentono alla soluzione di orchestrazione del contenitore di modificare l'ambiente di runtime senza ricostruire il contenitore. Questo modello include esempi di variabili di ambiente che possono essere utili per le applicazioni Blu Age.
- Convalida tutte le dipendenze dell'applicazione prima di eseguire l'applicazione Blu Age. Ad esempio, verificate che il database sia disponibile e che le credenziali siano valide. Scrivi dei test nello script wrapper per verificare le dipendenze e fallisci subito se non vengono soddisfatte.

- Usa la registrazione dettagliata all'interno dello script wrapper. Interagire direttamente con un container in esecuzione può essere difficile, a seconda della piattaforma di orchestrazione e della durata del lavoro. Assicurati che venga scritto un output utile per facilitare la diagnosi `STDOUT` di eventuali problemi. Ad esempio, l'output potrebbe includere il contenuto della directory di lavoro dell'applicazione sia prima che dopo l'esecuzione dell'applicazione.

Epiche

Ottieni un file JAR dell'applicazione Blu Age

Operazione	Description	Competenze richieste
Opzione 1: utilizza Blu Age per ottenere il file JAR dell'appl icazione.	Il contenitore in questo modello richiede un'applic azione Blu Age. In alternativa, è possibile utilizzare l'applica zione Java di esempio fornita con questo modello per un prototipo. Collaborate con il team di Blu Age per ottenere un file JAR per la vostra applicazi one che può essere inserito nel contenitore. Se il file JAR non è disponibile, consultate l'operazione successiva per utilizzare invece l'applicazione di esempio.	Architetto del cloud
Opzione 2: creare o utilizzare il file JAR dell'applicazione di esempio fornito.	Questo modello fornisce un file JAR di esempio predefini to. Questo file invia in output le variabili di ambiente dell'appl icazione `STDOUT` prima di dormire per 30 secondi e uscire.	Sviluppatore di app

Operazione	Description	Competenze richieste
	Questo file ha un nome `bluAgeSample.jar` e si trova nella cartella docker del GitHub repository. Se desideri modificare il codice e creare la tua versione del file JAR, usa il codice sorgente che si trova in. /java_sample/src/sample _java_app .java nel repository. GitHub È possibile utilizzare lo script di compilazione all'indirizzo. /java_sample/build.sh per compilare il codice sorgente Java e creare un nuovo file JAR.	

Crea il contenitore Blu Age

Operazione	Description	Competenze richieste
Clona il GitHub repository.	Clona il repository di codice di esempio utilizzando il comando: `git clone https://github.com/aws-samples/aws-blu-age-sample-container`	AWS DevOps
Usa Docker per creare il contenitore.	Usa Docker per creare il contenitore prima di inviarlo a un registro Docker come Amazon ECR:	AWS DevOps

Operazione	Description	Competenze richieste
	1. Dal terminale prescelto, accedi alla `docker` cartella all'interno del tuo repository locale. GitHub 2. Usa questo comando per creare il contenitore: `docker build -t <tag> .` `<tag>` dov'è il nome del contenitore che vuoi usare.	
Prova il contenitore Blu Age.	(Facoltativo) Se necessario, testate il contenitore localmente utilizzando il comando: `docker run -it <tag> /bin/bash`	AWS DevOps

Operazione	Description	Competenze richieste
Effettua l'autenticazione nel tuo repository Docker.	Se prevedi di utilizzare Amazon ECR, segui le istruzioni nella documentazione di Amazon ECR per installare e configurare l'AWS CLI e autenticare la CLI Docker nel tuo registro predefinito. Ti consigliamo di utilizzare il comando per l'autenticazione. get-login-password **Note** La console Amazon ECR fornisce una versione precompilata di questo comando se utilizzi il pulsante Visualizza comandi push. Per ulteriori informazioni, consulta la documentazione di Amazon ECR. `aws ecr get-login -password --region <region> | docker login --username AWS --password-stdin <account>.dkr.ecr.<region>.amazonaws.com`	AWS DevOps

Operazione	Description	Competenze richieste
	Se non intendi utilizzar e Amazon ECR, segui le istruzioni fornite per il tuo sistema di registro dei container.	
Crea un repository di contenito ri.	Crea un repository in Amazon ECR. Per istruzioni, consulta lo schema Distribuisci un ambiente per applicazioni Blu Age containerizzate utilizzan do Terraform. Se utilizzi un altro sistema di registro dei contenitori, segui le istruzioni fornite per quel sistema.	AWS DevOps

Operazione	Description	Competenze richieste

Etichetta e invia il contenitore al repository di destinazione.	Se utilizzi Amazon ECR:
	1. Etichetta l'immagine Docker locale con il registro e l'archivio Amazon ECR, in modo da poterla inviare al tuo repository remoto:
	`docker tag <tag>:latest <account>.dkr.ecr.<region>.amazonaws.com/<repository>:<versionNumber>`
	2. Invia l'immagine al repository remoto:
	`docker push <account>.dkr.ecr.<region>.amazonaws.com/<repository>:<versionNumber>`
	Per ulteriori informazioni, consulta Pushing a Docker image nella Amazon ECR User Guide.
		AWS DevOps

Risorse correlate

Risorse AWS

- Repository di contenitori di esempio di AWS Blu Age
- Esecuzione di carichi di lavoro mainframe Blu Age modernizzati su un'infrastruttura AWS serverless

- Implementa un ambiente per applicazioni Blu Age containerizzate utilizzando Terraform
- Utilizzo di Amazon ECR con l'AWS CLI (Amazon ECR User Guide)
- Autenticazione del registro privato (Amazon ECR User Guide)
- Documentazione Amazon ECS
- Documentazione Amazon EKS

Altre risorse

- Sito Web Blu Age
- Sito web Docker

Converti e decomprimi i dati EBCDIC in ASCII su AWS usando Python

Luis Gustavo Dantas, Amazon Web Services

Riepilogo

Poiché i mainframe in genere ospitano dati aziendali critici, la modernizzazione dei dati è una delle attività più importanti durante la migrazione dei dati verso il cloud Amazon Web Services (AWS) o un altro ambiente American Standard Code for Information Interchange (ASCII). Sui mainframe, i dati sono generalmente codificati in formato EBCDIC (Extended Binary-Coded Decimal Interchange Code). L'esportazione di database, Virtual Storage Access Method (VSAM) o file flat produce generalmente file EBCDIC binari compressi, la cui migrazione è più complessa. La soluzione di migrazione del database più utilizzata è Change Data Capture (CDC), che, nella maggior parte dei casi, converte automaticamente la codifica dei dati. Tuttavia, i meccanismi CDC potrebbero non essere disponibili per questi database, VSAM o file flat. Per questi file, è necessario un approccio alternativo per modernizzare i dati.

Questo modello descrive come modernizzare i dati EBCDIC convertendoli in formato ASCII. Dopo la conversione, puoi caricare i dati in database distribuiti o fare in modo che le applicazioni nel cloud elaborino direttamente i dati. Il pattern utilizza lo script di conversione e i file di esempio presenti nel mainframe-data-utilities GitHub repository.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un file di input EBCDIC e il corrispondente quaderno COBOL (Common Business-Oriented Language). Nel repository sono inclusi un file EBCDIC di esempio e un quaderno COBOL. mainframe-data-utilities GitHub Per ulteriori informazioni sui quaderni COBOL, consulta Enterprise COBOL for 6.4 Programming Guide sul sito Web IBM. z/OS

Limitazioni

- I layout di file definiti all'interno dei programmi COBOL non sono supportati. Devono essere resi disponibili separatamente.

Versioni del prodotto

- Python versione 3.8 o successiva

Architecture

Stack tecnologico di origine

- Dati EBCDIC su un mainframe
- Quaderno COBOL

Stack tecnologico Target

- Istanza Amazon Elastic Compute Cloud (Amazon EC2) in un cloud privato virtuale (VPC)
- Amazon Elastic Block Store (Amazon EBS)
- Python e i suoi pacchetti richiesti, JavaScript Object Notation (JSON), sys e datetime
- File flat ASCII pronto per essere letto da un'applicazione moderna o caricato in una tabella di database relazionale

Architettura Target

Il diagramma di architettura mostra il processo di conversione di un file EBCDIC in un file ASCII su un'istanza: EC2

1. Utilizzando lo script parse_copybook_to_json.py, convertite il quaderno COBOL in un file JSON.
2. Utilizzando il file JSON e lo script extract_ebcdic_to_ascii.py, convertite i dati EBCDIC in un file ASCII.

Automazione e scalabilità

Dopo aver predisposto le risorse necessarie per le prime conversioni manuali dei file, puoi automatizzare la conversione dei file. Questo modello non include istruzioni per l'automazione. Esistono diversi modi per automatizzare la conversione. Di seguito è riportata una panoramica di un approccio possibile:

1. Incapsula i comandi di script AWS Command Line Interface (AWS CLI) e Python in uno script di shell.

2. Crea una funzione AWS Lambda che invii in modo asincrono il job dello script di shell a un'istanza. EC2 Per ulteriori informazioni, consulta Pianificazione di lavori SSH con AWS Lambda.
3. Crea un trigger Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) che richiami la funzione Lambda ogni volta che viene caricato un file legacy. Per ulteriori informazioni, consulta Usare un trigger Amazon S3 per richiamare una funzione Lambda.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e scalarli rapidamente verso l'alto o verso il basso.
- Amazon Elastic Block Store (Amazon EBS) fornisce volumi di storage a livello di blocco da utilizzare con istanze Amazon Elastic Compute Cloud (Amazon). EC2
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.

Altri strumenti

- GitHubè un servizio di code-hosting che fornisce strumenti di collaborazione e controllo delle versioni.
- Python è un linguaggio di programmazione di alto livello.

Archivio di codice

Il codice per questo pattern è disponibile nel mainframe-data-utilities GitHub repository.

Epiche

Prepara l'istanza EC2

Operazione	Description	Competenze richieste
Avvia un' EC2 istanza.	L' EC2 istanza deve disporre di accesso a Internet in uscita. Ciò consente all'istanza di accedere al codice sorgente Python disponibile su. GitHub Per creare l'istanza: 1. Apri la EC2 console Amazon in https://console.aws.amazon.com/ec2. 2. Avvia un'istanza EC2 Linux. Utilizza un indirizzo IP pubblico e consenti l'accesso in entrata tramite la porta 22. Assicurati che la dimensione di archiviazione dell'istanza sia almeno il doppio della dimensione del file di dati EBCDIC. Per istruzioni, consulta la EC2 documentazione di Amazon.	Informazioni generali su AWS
Installa Git.	1. Utilizzando un client Secure Shell (SSH), connettiti all' EC2 istanza che hai appena lanciato. Per ulteriori informazioni, consulta Connect to your Linux istance.	Informazioni generali su AWS, Linux

Operazione	Description	Competenze richieste

	2. Nella EC2 console Amazon, esegui il seguente comando. Questo installa Git sull' EC2 istanza.
	`sudo yum install git`
	3. Esegui il comando seguente e conferma che Git è stato installato correttamente.
	`git --version`	

Operazione	Description	Competenze richieste

Installare Python.	1. Nella EC2 console Amazon, esegui il seguente comando. Questo installa Python sull' EC2 istanza.
	`sudo yum install python3`
	2. Nella EC2 console Amazon, esegui il seguente comando. Questo installa Pip3 sull'istanza. EC2
	`sudo yum install python3-pip`
	3. Nella EC2 console Amazon, esegui il seguente comando. Questo installa AWS SDK per Python (Boto3) sull'istanza. EC2
	`sudo pip3 install boto3`
	4. Nella EC2 console Amazon, esegui il seguente comando, `<us-east-1>` dov'è il codice per la tua regione AWS. Per un elenco completo dei codici regionali, consulta la sezione Regioni disponibili nella EC2 documentazione di Amazon.
		Informazioni generali su AWS, Linux

Operazione	Description	Competenze richieste
	`export AWS_DEFAU LT_REGION=<us-east -1>`	
Clona il GitHub repository.	1. Nella EC2 console Amazon, esegui il seguente comando. Questo clona il mainframe-data-uti litiesrepository da GitHub e apre la posizione di copia predefinita, la home cartella. `git clone https://g ithub.com/aws-samp les/mainframe-data- utilities.git` 2. Nella home cartella, verifica che la `mainframe-data-utilities` cartella sia presente.	AWS generale, GitHub

Crea il file ASCII dai dati EBCDIC

Operazione	Description	Competenze richieste
Analizza il quaderno COBOL nel file di layout JSON.	All'interno della **`mainframe-data-utilities`** cartella, esegui lo script parse_cop ybook_to_json.py. Questo modulo di automazione legge il layout del file da un quaderno COBOL e crea un file JSON. Il file JSON contiene le informazioni	Informazioni generali su AWS, Linux

Operazione	Description	Competenze richieste

	necessarie per interpretare ed estrarre i dati dal file sorgente. Questo crea i metadati JSON dal quaderno COBOL.
	
	Il comando seguente converte il quaderno COBOL in un file JSON.
	
	`python3 parse_copybook_to_json.py \ -copybook LegacyReference/COBPACK2.cpy \ -output sample-data/cobpack2-list.json \ -dict sample-data/cobpack2-dict.json \ -ebcdic sample-data/COBPACK.OUTFILE.txt \ -ascii sample-data/COBPACK.ASCII.txt \ -print 10000`
	
	Lo script stampa gli argomenti ricevuti.
	
	`---------- ---------- ---------- ---------- Copybook file..... .....| LegacyReference/COBPACK2.cpy Parsed copybook (JSON List).| sample-data/cobpack2-list.json JSON Dict (documentation)...| sample-data/cobpack2-dict.json`	

Operazione	Description	Competenze richieste

	<pre>ASCII file.....
	.....| sample-
	data/COBPACK.ASCII.t
	xt
	EBCDIC file.....
	.....| sample-
	data/COBPACK.OUTFILE
	.txt
	Print each.....
	.....| 10000
	---------- 	
	---------- 	
	---------- 	
	----------</pre> Per ulteriori informazioni sugli argomenti, consultate il file README nel GitHub repository.	

Operazione	Description	Competenze richieste
Ispeziona il file di layout JSON.	1. Passate al percorso di output definito nello script parse_copybook_to_json.py. 2. Controlla l'ora di creazione del file sample-data/cobpack2-list.json per confermare di aver selezionato il file di layout JSON appropriato. 3. Esamina il file JSON e verifica che il contenuto sia simile al seguente. `"input": "extract-ebcdic-to-ascii/COBPACK.OUTFILE.txt", "output": "extract-ebcdic-to-ascii/COBPACK.ASCII.txt", "max": 0, "skip": 0, "print": 10000, "lrecl": 150, "rem-low-values": true, "separator": "|", "transf": [ { "type": "ch", "bytes": 19, "name": "OUTFILE-TEXT" }` Gli attributi più importanti del file di layout JSON sono:	Informazioni generali su AWS, JSON

Operazione	Description	Competenze richieste
	• `input`— Contiene il percorso del file EBCDIC da convertire • `output`— Definisce il percorso in cui verrà generato il file ASCII • `lrecl`— specifica la dimensione in byte della lunghezza del record logico • `transf`— Elenca tutti i campi e le relative dimensioni in byte Per ulteriori informazioni sul file di layout JSON, consultate il file README nel repository. GitHub	

Operazione	Description	Competenze richieste

Crea il file ASCII.	Esegui lo script extract_e bcdic_to_ascii.py, incluso nel repository GitHub clonato. Questo script legge il file EBCDIC e scrive un file ASCII convertito e leggibile.
	`python3 extract_e bcdic_to_ascii.py - local-json sample-da ta/cobpack2-list.json`
	Durante l'elaborazione dei dati EBCDIC, lo script stampa un messaggio per ogni batch di 10.000 record. Guarda l'esempio seguente.
	`---------- ---------- ---------- ---------- 2023-05-15 21:21:46. 322253 | Local Json file | -local-json | sample-data/cobpack2- list.json 2023-05-15 21:21:47. 034556 | Records processed | 10000 2023-05-15 21:21:47. 736434 | Records processed | 20000 2023-05-15 21:21:48. 441696 | Records processed | 30000`
		Informazioni generali su AWS

Operazione	Description	Competenze richieste
	2023-05-15 21:21:49. 173781 | Records processed | 40000 2023-05-15 21:21:49. 874779 | Records processed | 50000 2023-05-15 21:21:50. 705873 | Records processed | 60000 2023-05-15 21:21:51. 609335 | Records processed | 70000 2023-05-15 21:21:52. 292989 | Records processed | 80000 2023-05-15 21:21:52. 938366 | Records processed | 89280 2023-05-15 21:21:52. 938448 Seconds 6.616232 Per informazioni su come modificare la frequenza di stampa, consultate il file README nel repository. GitHub	

Operazione	Description	Competenze richieste

Esamina il file ASCII.	1. Controlla l'ora di creazione del file extract-ebcdic-to-ascii/CobPack.ASCII.txt per verificare che sia stato creato di recente.
	2. Nella EC2 console Amazon, inserisci il seguente comando. Questo apre il primo record del file ASCII.
	`head sample-data/COBPACK.ASCII.txt -n 1| xxd`
	3. Esamina il contenuto del primo record. Poiché i file EBCDIC sono generalmente binari, non hanno caratteri speciali Carriage Return and Line Feed (CRLF). Lo script extract_ebcdic_to_ascii.py aggiunge un carattere pipe come separatore di colonna, definito nei parametri dello script.
	
	Se è stato utilizzato il file EBCDIC di esempio fornito, il seguente è il primo record del file ASCII.
	`00000000: 2d30 3030 3030 3030 3030 3130 3030 3030  -000000000100000`
		Informazioni generali su AWS, Linux

Operazione	Description	Competenze richieste
	00000010: 3030 307c 3030 3030 3030 3030 3031 3030 000|00000 0000100 00000020: 3030 3030 3030 7c2d 3030 3030 3030 3030 000000|-0 0000000 00000030: 3031 3030 3030 3030 3030 7c30 7c30 7c31 0100000000| 0|0|1 00000040: 3030 3030 3030 3030 7c2d 3130 3030 3030 00000000| -100000 00000050: 3030 307c 3130 3030 3030 3030 307c 2d31 000|10000 0000|-1 00000060: 3030 3030 3030 3030 7c30 3030 3030 7c30 00000000| 00000|0 00000070: 3030 3030 7c31 3030 3030 3030 3030 7c2d 0000|1000 00000|- 00000080: 3130 3030 3030 3030 307c 3030 3030 3030 100000000| 000000 00000090: 3030 3030 3130 3030 3030 3030 307c 2d30 000010000 0000|-0 000000a0: 3030 3030 3030 3030 3031 3030 3030 3030 000000000 1000000	

Operazione	Description	Competenze richieste
	000000b0: 3030 7c41 7c41 7c0a 00|A|A|.	

Operazione	Description	Competenze richieste
Valuta il file EBCDIC.	Nella EC2 console Amazon, inserisci il seguente comando. Questo apre il primo record del file EBCDIC. `head sample-data/COBPACK.OUTFILE.txt -c 150 | xxd` Se hai utilizzato il file EBCDIC di esempio, il risultato è il seguente. `00000000: 60f0 f0f0 f0f0 f0f0 f0f0 f0f0 f1f0 f0f0 f0f0 `........ ....` `00000010: f0f0 f0f0 f0f0 f0f0 f0f0 f0f0 f1f0 f0f0 ....` `00000020: f0f0 f0f0 f0f0 f0f0 f0f0 f0f0 f1f0 ....` `00000030: f0f0 f0f0 f0f0 d000 0000 0005 f5e1 00fa ....` `00000040: 0a1f 0000 0000 0005 f5e1 00ff ffff fffa ....` `00000050: 0a1f 0000 000f 0000 0c10 0000 000f 1000 ....` `00000060: 0000 0d00 0000 0000 1000 0000`	Informazioni generali su AWS, Linux, EBCDIC

Operazione	Description	Competenze richieste
	0f00 0000 .... 00000070: 0000 1000 0000 0dc1 c100 0000 0000 0000 .... 00000080: 0000 0000 0000 0000 0000 0000 0000 0000 .... 00000090: 0000 0000 0000	
	Per valutare l'equivalenza tra il file di origine e quello di destinazione, è richiesta una conoscenza approfondita di EBCDIC. Ad esempio, il primo carattere del file EBCDIC di esempio è un trattino (). - Nella notazione esadecima le del file EBCDIC, questo carattere è rappresentato da, e nella notazione esadecima le del file ASCII60, questo carattere è rappresentato da. 2D Per una tabella di EBCDIC-to-ASCII conversio ne, consulta EBCDIC to ASCII sul sito Web IBM.	

Risorse correlate

Riferimenti

- Il set di caratteri EBCDIC (documentazione IBM)

- Da EBCDIC ad ASCII (documentazione IBM)
- COBOL (documentazione IBM)
- Concetti JCL di base (documentazione IBM)
- Connect alla tua istanza Linux (EC2 documentazione Amazon)

Tutorial

- Pianificazione di lavori SSH con AWS Lambda (post sul blog AWS)
- Utilizzo di un trigger Amazon S3 per richiamare una funzione Lambda (documentazione AWS Lambda)

Converti i file mainframe dal formato EBCDIC al formato ASCII delimitato da caratteri in Amazon S3 utilizzando AWS Lambda

Luis Gustavo Dantas, Amazon Web Services

Riepilogo

Questo modello mostra come avviare una AWS Lambda funzione che converte automaticamente i file EBCDIC (Extended Binary Coded Decimal Interchange Code) del mainframe in file ASCII (American Standard Code for Information Interchange) delimitati da caratteri. La funzione Lambda viene eseguita dopo il caricamento dei file ASCII in un bucket Amazon Simple Storage Service (Amazon S3). Dopo la conversione dei file, puoi leggere i file ASCII su carichi di lavoro basati su x86 o caricare i file in database moderni.

L'approccio alla conversione dei file illustrato in questo modello può aiutarvi a superare le sfide legate all'utilizzo dei file EBCDIC in ambienti moderni. I file codificati in EBCDIC contengono spesso dati rappresentati in formato binario o decimale compresso e i campi sono a lunghezza fissa. Queste caratteristiche creano ostacoli perché i carichi di lavoro o gli ambienti distribuiti moderni basati su x86 generalmente funzionano con dati con codifica ASCII e non sono in grado di elaborare file EBCDIC.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un bucket Amazon S3
- Un utente AWS Identity and Access Management (IAM) con autorizzazioni amministrative
- AWS CloudShell
- Python 3.8.0 o successivo
- Un file flat codificato in EBCDIC e la struttura dati corrispondente in un quaderno COBOL (Common Business-Oriented Language)

> **Note**
>
> Questo modello utilizza un file EBCDIC di esempio (Client.EBCDIC.txt) e il quaderno COBOL corrispondente (COBKS05.cpy). Entrambi i GitHub mainframe-data-utilitiesfile sono disponibili nel repository.

Limitazioni

- I quaderni COBOL di solito contengono più definizioni di layout. Il mainframe-data-utilitiesprogetto può analizzare questo tipo di quaderno ma non può dedurre quale layout prendere in considerazione per la conversione dei dati. Questo perché i quaderni non seguono questa logica (che rimane invece nei programmi COBOL). Di conseguenza, è necessario configurare manualmente le regole per la selezione dei layout dopo aver analizzato il quaderno.
- Questo modello è soggetto alle quote Lambda.

Architecture

Stack tecnologico di origine

- IBM z/OS, IBM i e altri sistemi EBCDIC
- File sequenziali con dati codificati in EBCDIC (come IBM Db2 unloads)
- Quaderno COBOL

Stack tecnologico Target

- Simple Storage Service (Amazon S3)
- Notifica degli eventi di Amazon S3
- IAM
- funzione Lambda
- Python 3.8 o versioni successive
- Utilità per dati mainframe
- Metadati JSON
- File ASCII delimitati da caratteri

Architettura di Target

Il diagramma seguente mostra un'architettura per la conversione dei file EBCDIC mainframe in file ASCII.

Il diagramma mostra il flusso di lavoro seguente:

1. L'utente esegue lo script di analisi dei copybook, che converte il quaderno COBOL in un file JSON.
2. L'utente carica i metadati JSON in un bucket Amazon S3. Ciò rende i metadati leggibili dalla funzione Lambda di conversione dei dati.
3. L'utente o un processo automatizzato carica il file EBCDIC nel bucket Amazon S3.
4. L'evento di notifica di Amazon S3 attiva la funzione Lambda di conversione dei dati.
5. AWS verifica le autorizzazioni di lettura/scrittura del bucket Amazon S3 per la funzione Lambda.
6. Lambda legge il file dal bucket Amazon S3 e lo converte localmente da EBCDIC in ASCII.
7. Lambda registra lo stato del processo in Amazon. CloudWatch
8. Lambda riscrive il file ASCII su Amazon S3.

> **Note**
>
> Lo script di analisi dei copybook viene eseguito una sola volta per eseguire la conversione dei metadati in formato JSON, che viene successivamente archiviato in un bucket Amazon S3. Dopo la conversione iniziale, tutti i file EBCDIC successivi che fanno riferimento allo stesso file JSON nel bucket Amazon S3 utilizzeranno la configurazione dei metadati esistente.

Tools (Strumenti)

Servizi AWS

- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue AWS risorse e delle applicazioni su cui esegui AWS in tempo reale.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS CloudShellè una shell basata su browser che puoi utilizzare per gestire Servizi AWS utilizzando AWS Command Line Interface (AWS CLI) e una gamma di strumenti di sviluppo preinstallati.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Lambda esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.

Altri strumenti

- GitHubè un servizio di code-hosting che fornisce strumenti di collaborazione e controllo delle versioni.
- Python è un linguaggio di programmazione di alto livello.

Codice

Il codice per questo pattern è disponibile nel repository. GitHub mainframe-data-utilities

Best practice

Prendi in considerazione le seguenti best practice:

- Imposta le autorizzazioni richieste a livello di Amazon Resource Name (ARN).
- Concedi sempre le autorizzazioni con privilegi minimi per le policy IAM. Per ulteriori informazioni, consulta le best practice di sicurezza in IAM nella documentazione IAM.

Epiche

Crea variabili di ambiente e una cartella di lavoro

Operazione	Description	Competenze richieste

Creare le variabili di ambiente.	Copiate le seguenti variabili di ambiente in un editor di testo, quindi sostituite i `<placeholder>` valori dell'esempio seguente con i valori delle risorse:
	`bucket=<your_bucket_name>`
	`account=<your_account_number>`
	`region=<your_region_code>`
		Informazioni generali su AWS

Operazione	Description	Competenze richieste
	Note Creerai riferimenti al tuo bucket Amazon S3 e versioni successiv e Account AWS. Regione AWS Per definire le variabili di ambiente, apri la CloudShel l console, quindi copia e incolla le variabili di ambiente aggiornate nella riga di comando. **Note** È necessario ripetere questo passaggio ogni volta che la CloudShel l sessione viene riavviata.	

Operazione	Description	Competenze richieste
Crea una cartella di lavoro.	Per semplificare il processo di pulizia delle risorse in un secondo momento, crea una cartella di lavoro CloudShell eseguendo il seguente comando: `mkdir workdir; cd workdir` **Note** È necessario modificare la directory nella directory di lavoro (`workdir`) ogni volta che si perde la connessione alla CloudShell sessione.	Informazioni generali su AWS

Definisci un ruolo e una policy IAM

Operazione	Description	Competenze richieste
Crea una politica di fiducia per la funzione Lambda.	Il convertitore EBCDIC funziona con una funzione Lambda. La funzione deve avere un ruolo IAM. Prima di creare il ruolo IAM, è necessario definire un documento sulla politica di fiducia che consenta alle	Informazioni generali su AWS

Operazione	Description	Competenze richieste

	risorse di assumere tale politica.
	
	Dalla cartella CloudShell di lavoro, crea un documento di policy eseguendo il seguente comando:
	
	`E2ATrustPol=$(cat <<EOF`
	`{`
	`"Version": "2012-10-17",`
	`"Statement": [`
	`{`
	`"Effect": "Allow",`
	`"Principal": {`
	`"Service": "lambda.amazonaws.com"`
	`},`
	`"Action": "sts:AssumeRole"`
	`}`
	`]`
	`}`
	`EOF`
	`)`
	`printf "$E2ATrustPol" > E2ATrustPol.json`	

Operazione	Description	Competenze richieste
Crea il ruolo IAM per la conversione Lambda.	Per creare un ruolo IAM, esegui il seguente AWS CLI comando dalla cartella di CloudShell lavoro: `aws iam create-role --role-name E2AConvLambdaRole --assume-role-policy-document file://E2ATrustPol.json`	Informazioni generali su AWS

Operazione	Description	Competenze richieste

Crea il documento di policy IAM per la funzione Lambda.	La funzione Lambda deve disporre dell'accesso in lettura/scrittura al bucket Amazon S3 e delle autorizzazioni di scrittura per Amazon Logs. CloudWatch
	
	Per creare una policy IAM, esegui il seguente comando dalla cartella di lavoro: CloudShell
	
	`E2APolicy=$(cat <<EOF`
	`{`
	`"Version": "2012-10-17",`
	`"Statement": [`
	`{`
	`"Sid": "Logs",`
	`"Effect": "Allow",`
	`"Action": [`
	`"logs:PutLogEvents",`
	`"logs:CreateLogStream",`
	`"logs:CreateLogGroup"`
	`],`
	`"Resource": [`
	`"arn:aws:logs:*:*:log-group:*",`
	`"arn:aws:logs:*:*:`
		Informazioni generali su AWS

Operazione	Description	Competenze richieste

	log-group:*:log-st
	ream:*"
	            ]
	        },
	        {
	            "Sid": "S3",
	            "Effect":
	 "Allow",
	            "Action": [
	
	 "s3:GetObject",
	
	 "s3:PutObject",
	
	 "s3:GetObjectVersion"
	            ],
	            "Resource":
	 [
	
	 "arn:aws:s3:::%s/*",
	
	 "arn:aws:s3:::%s"
	            ]
	        }
	    ]
	}
	EOF
	)
	printf "$E2APolicy"
	 "$bucket" "$bucket"
	 > E2AConvLambdaPolic
	y.json	

Operazione	Description	Competenze richieste
Allega il documento relativo alla policy IAM al ruolo IAM.	Per allegare la policy IAM al ruolo IAM, inserisci il seguente comando dalla tua cartella di CloudShell lavoro: `aws iam put-role-policy --role-name E2AConvLambdaRole --policy-name E2AConvLambdaPolicy --policy-document file://E2AConvLambdaPolicy.json`	Informazioni generali su AWS

Crea la funzione Lambda per la conversione EBCDIC

Operazione	Description	Competenze richieste
Scarica il codice sorgente della conversione EBCDIC.	Dalla cartella CloudShell di lavoro, esegui il seguente comando per scaricare il codice mainframe-data-utilities sorgente da: GitHub `git clone https://github.com/aws-samples/mainframe-data-utilities.git mdu`	Informazioni generali su AWS
Crea il pacchetto ZIP.	Dalla cartella di CloudShell lavoro, immettete il seguente comando per creare il pacchetto ZIP che crea la funzione Lambda per la conversione EBCDIC:	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	`cd mdu; zip ../mdu.zip *.py; cd ..`	
Creazione della funzione Lambda	Dalla cartella di CloudShell lavoro, immettete il seguente comando per creare la funzione Lambda per la conversione EBCDIC: `aws lambda create-function \ --function-name E2A \ --runtime python3.9 \ --zip-file fileb://mdu.zip \ --handler extract_ebcdic_to_ascii.lambda_handler \ --role arn:aws:iam::$account:role/E2AConvLambdaRole \ --timeout 10 \ --environment "Variables={layout=$bucket/layout/}"` **Note** Il layout della variabile di ambiente indica alla funzione Lambda dove risiedono i metadati JSON.	Informazioni generali su AWS

Operazione	Description	Competenze richieste

Crea la politica basata sulle risorse per la funzione Lambda.	Dalla cartella di CloudShell lavoro, inserisci il seguente comando per consentire alla notifica degli eventi di Amazon S3 di attivare la funzione Lambda per la conversione EBCDIC:
	`aws lambda add-permission \`
	`--function-name E2A \`
	`--action lambda:InvokeFunction \`
	`--principal s3.amazonaws.com \`
	`--source-arn arn:aws:s3:::$bucket \`
	`--source-account $account \`
	`--statement-id 1`
		Informazioni generali su AWS

Crea la notifica degli eventi di Amazon S3

Operazione	Description	Competenze richieste
Crea il documento di configurazione per la notifica degli eventi di Amazon S3.	La notifica degli eventi di Amazon S3 avvia la funzione Lambda di conversione EBCDIC quando i file vengono inseriti nella cartella di input. Dalla cartella di CloudShell lavoro, esegui il seguente comando per creare il documento JSON per la	Informazioni generali su AWS

Operazione	Description	Competenze richieste

	notifica degli eventi di Amazon S3:
	
	`S3E2AEvent=$(cat <<EOF`
	`{`
	`"LambdaFunctionConf`
	`igurations": [`
	`{`
	`"Id": "E2A",`
	`"LambdaFu`
	`nctionArn": "arn:aws:`
	`lambda:%s:%s:funct`
	`ion:E2A",`
	`"Events":`
	`[ "s3:ObjectCreated:`
	`Put" ],`
	`"Filter": {`
	`"Key": {`
	`"FilterRu`
	`les": [`
	`{`
	`"Name":`
	`"prefix",`
	`"Value":`
	`"input/"`
	`}`
	`]`
	`}`
	`}`
	`}`
	`]`
	`}`
	`EOF`
	`)`
	`printf "$S3E2AEvent"`
	`"$region" "$account" >`
	`S3E2AEvent.json`	

Operazione	Description	Competenze richieste
Crea la notifica degli eventi di Amazon S3.	Dalla cartella di CloudShell lavoro, inserisci il seguente comando per creare la notifica degli eventi di Amazon S3: `aws s3api put-bucket-notification-configuration --bucket $bucket --notification-configuration file://S3E2AEvent.json`	Informazioni generali su AWS

Crea e carica i metadati JSON

Operazione	Description	Competenze richieste

Analizza il quaderno COBOL.	Dalla cartella di CloudShell lavoro, immettete il seguente comando per analizzare un quaderno COBOL di esempio in un file JSON (che definisce come leggere e suddividere correttamente il file di dati):
	`python3 mdu/parse_copybook_to_json.py \`
	`-copybook mdu/LegacyReference/COBKS05.cpy \`
	`-output CLIENT.json \`
	`-output-s3key CLIENT.ASCII.txt \`
	`-output-s3bkt $bucket \`
	`-output-type s3 \`
		Informazioni generali su AWS

Operazione	Description	Competenze richieste
	`-print 25`	

Aggiungi la regola di trasformazione.	Il file di dati di esempio e il corrispondente quaderno COBOL sono un file a più layout. Ciò significa che la conversione deve suddividere i dati in base a determinate regole. In questo caso, i byte nelle posizioni 3 e 4 in ogni riga definiscono il layout.
	
	Dalla cartella CloudShell di lavoro, modificate il `CLIENT.json` file e modificate il contenuto nel `"transf-rule": [],` modo seguente:
	
	`"transf-rule": [`
	`{`
	`"offset": 4,`
	`"size": 2,`
	`"hex": "0002",`
	`"transf": "transf1"`
	`},`
	`{`
	`"offset": 4,`
	`"size": 2,`
	`"hex": "0000",`
	`"transf": "transf2"`
	`}`
	`],`
		Informazioni generali su AWS, mainframe IBM, Cobol

Operazione	Description	Competenze richieste
Carica i metadati JSON nel bucket Amazon S3.	Dalla cartella di CloudShell lavoro, inserisci il seguente AWS CLI comando per caricare i metadati JSON nel tuo bucket Amazon S3: `aws s3 cp CLIENT.json s3://$bucket/layout/CLIENT.json`	Informazioni generali su AWS

Convertire il file EBCDIC

Operazione	Description	Competenze richieste
Invia il file EBCDIC al bucket Amazon S3.	Dalla cartella di CloudShell lavoro, inserisci il seguente comando per inviare il file EBCDIC al bucket Amazon S3: `aws s3 cp mdu/sample-data/CLIENT.EBCDIC.txt s3://$bucket/input/` **Note** Ti consigliamo di impostare cartelle diverse per i file di input (EBCDIC) e di output (ASCII) per evitare di richiamar e nuovamente la funzione di conversio	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	ne Lambda quando il file ASCII viene caricato nel bucket Amazon S3.	

Operazione	Description	Competenze richieste

Controlla l'output.	Dalla cartella di CloudShell lavoro, inserisci il seguente comando per verificare se il file ASCII è generato nel tuo bucket Amazon S3:
	`aws s3 ls s3://$bucket/`
	Note
	La conversione dei dati può richiedere diversi secondi. Ti consigliamo di verificare la presenza del file ASCII alcune volte.
	Una volta che il file ASCII è disponibile, inserisci il seguente comando per visualizzare il contenuto del file convertito nel bucket Amazon S3. Se necessario, puoi scaricarlo o utilizzarlo direttamente dal bucket Amazon S3:
	`aws s3 cp s3://$bucket/ CLIENT.ASCII.txt - | head`
	Controlla il contenuto del file ASCII:
		Informazioni generali su AWS

Operazione	Description	Competenze richieste

	`0|0|220|`
	`1|1|HERBERT MOHAMED|1`
	`958-08-31|BACHELOR|`
	`0010000.00|`
	`1|2|36|THE ROE AVENUE|`
	`2|1|JAYLEN GEORGE|19`
	`69-05-29|ELEMENTARY|`
	`0020000.00|`
	`2|2|365|HEATHFIELD`
	` ESPLANADE|`
	`3|1|MIKAEEL WEBER|198`
	`2-02-17|MASTER|003`
	`0000.00|`
	`3|2|4555|MORRISON`
	` STRAND|`
	`4|1|APRIL BARRERA|1`
	`967-01-12|DOCTOR|0`
	`030000.00|`
	`4|2|1311|MARMION PARK|`
	`5|1|ALEEZA PLANT|198`
	`5-03-01|BACHELOR|0`
	`008000.00|`	

Pulisci l'ambiente

Operazione	Description	Competenze richieste
(Facoltativo) Prepara le variabili e la cartella.	Se perdete la connessione con CloudShell, riconnettetevi e immettete il seguente comando per spostare la directory nella cartella di lavoro: `cd workdir`	Informazioni generali su AWS

Operazione	Description	Competenze richieste

	Assicuratevi che le variabili di ambiente siano definite:
	`bucket=<your_bucket_name>`
	`account=<your_account_number>`
	`region=<your_region_code>`	

Rimuovi la configurazione di notifica per il bucket.	Dalla cartella di CloudShell lavoro, esegui il seguente comando per rimuovere la configurazione di notifica degli eventi di Amazon S3:
	`aws s3api put-bucket-notification-configuration \`
	`--bucket=$bucket \`
	`--notification-configuration="{}"`
		Informazioni generali su AWS

Elimina la funzione Lambda.	Dalla cartella di CloudShell lavoro, immettete il seguente comando per eliminare la funzione Lambda per il convertitore EBCDIC:
	`aws lambda delete-function \`
	`--function-name E2A`
		Informazioni generali su AWS

Operazione	Description	Competenze richieste

Elimina il ruolo e la policy IAM.	Dalla cartella di CloudShell lavoro, inserisci il seguente comando per rimuovere il ruolo e la politica del convertit ore EBCDIC:
	`aws iam delete-role-policy \`
	`--role-name E2AConvLa mbdaRole \`
	`--policy-name E2AConvLa mbdaPolicy`
	
	`aws iam delete-role \`
	`--role-name E2AConvLa mbdaRole`
		Informazioni generali su AWS

Elimina i file generati nel bucket Amazon S3.	Dalla cartella di CloudShell lavoro, inserisci il seguente comando per eliminare i file generati nel bucket Amazon S3:
	`aws s3 rm s3://$bucket/layout --recursive`
	`aws s3 rm s3://$bucket/input --recursive`
	`aws s3 rm s3://$bucket/CLIENT.ASCII.txt`
		Informazioni generali su AWS

Elimina la cartella di lavoro.	Dalla cartella di CloudShell lavoro, immettete il seguente comando per `workdir` rimuoverne il contenuto: `cd ..; rm -Rf workdir`	Informazioni generali su AWS

Risorse correlate

- Utilità per dati mainframe README () GitHub
- Il set di caratteri EBCDIC (documentazione IBM)
- Da EBCDIC ad ASCII (documentazione IBM)
- COBOL (documentazione IBM)
- Utilizzo di un trigger Amazon S3 per richiamare una funzione Lambda (documentazione)AWS Lambda

Convertite file di dati mainframe con layout di registrazione complessi utilizzando Micro Focus

Peter West, Amazon Web Services

Riepilogo

Nota: Modernizzazione del mainframe AWS Service (Managed Runtime Environment experience) non è più aperto a nuovi clienti. Per funzionalità simili a Modernizzazione del mainframe AWS Service (Managed Runtime Environment experience), esplora Modernizzazione del mainframe AWS Service (Self-Managed Experience). I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta Modifica della Modernizzazione del mainframe AWS disponibilità.

Questo modello mostra come convertire i file di dati mainframe con dati non testuali e layout di record complessi dalla codifica dei caratteri EBCDIC (Extended Binary Coded Decimal Interchange Code) alla codifica dei caratteri ASCII (American Standard Code for Information Interchange) utilizzando un file di struttura Micro Focus. Per completare la conversione del file, è necessario effettuare le seguenti operazioni:

1. Preparate un unico file sorgente che descriva tutti gli elementi di dati e i layout dei record nell'ambiente mainframe.
2. Create un file di struttura che contenga il layout di registrazione dei dati utilizzando Micro Focus Data File Editor come parte di Micro Focus Classic Data File Tools o Data File Tools. Il file di struttura identifica i dati non testuali in modo da poter convertire correttamente i file mainframe da EBCDIC in ASCII.
3. Verificate il file di struttura utilizzando Classic Data File Tools o Data File Tools.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Micro Focus Enterprise Developer per Windows, disponibile tramite AWS Mainframe Modernization

Versioni del prodotto

- Micro Focus Enterprise Server 7.0 e versioni successive

Tools (Strumenti)

- Micro Focus Enterprise Developer fornisce l'ambiente di esecuzione per le applicazioni create con qualsiasi variante dell'ambiente di sviluppo integrato (IDE) di Enterprise Developer.
- Micro Focus Classic Data File Tools vi aiuta a convertire, navigare, modificare e creare file di dati. I Classic Data File Tools includono Data File Converter, Record Layout Editor e Data File Editor.
- Micro Focus Data File Tools vi aiuta a creare, modificare e spostare file di dati. I Data File Tools includono Data File Editor, File Conversion Utilities e Data File Structure Command Line Utility.

Epiche

Prepara il file sorgente

Operazione	Description	Competenze richieste
Identifica i componenti di origine.	Identifica tutti i possibili layout di record per il file, incluse eventuali ridefinizioni che contengono dati non testuali. Se disponete di layout che contengono ridefinizioni, dovete suddividerli in layout unici che descrivano ogni possibile permutazione della struttura dei dati. In genere, i layout dei record di un file di dati possono essere descritti dai seguenti archetipi: • Layout di registrazione con solo dati di testo • Registra il layout con dati non testuali • Layout di registrazione con dati non testuali subordinati a una clausola REDEFINES	Sviluppatore di app

Operazione	Description	Competenze richieste
	Per ulteriori informazioni sulla creazione di layout di record semplificati per file che contengono layout di record complessi, vedere Rehosting di applicazioni EBCDIC su ambienti ASCII per le migrazioni mainframe.	

Operazione	Description	Competenze richieste

Identifica le condizioni di layout dei record.	Per i file con più layout di record o i file che contengon o layout complessi con una clausola REDEFINES, identific ate i dati e le condizioni all'interno di un record che potete utilizzare per definire il layout da utilizzare durante la conversione. Si consiglia di discutere di questa attività con un esperto in materia (SME) che conosca i programmi che elaborano questi file.
	
	Ad esempio, un file può contenere due tipi di record che contengono dati non testuali. È possibile controlla re il codice sorgente ed eventualmente trovare codice simile al seguente:
	
	`MOVE "M" TO PART-TYPE`
	`MOVE "MAIN ASSEMBLY" TO`
	`PART-NAME`
	`MOVE "S" TO PART-TYPE`
	`MOVE "SUB ASSEMBLY 1"`
	`TO PART-NAME`
	
	Il codice consente di identific are quanto segue:
	
	• Il campo «PART-TYP E» viene utilizzato per determinare il tipo di record
		Sviluppatore di app

Operazione	Description	Competenze richieste
	• Il valore «M» viene utilizzato per «M-PART-RECORD» • Il valore «S» viene utilizzato per «S-PART-RECORD» È possibile documentare i valori utilizzati da questo campo per associare i layout dei record ai record di dati corretti nel file.	
Crea il file sorgente.	Se il file è descritto in più file di origine o se il layout del record contiene dati non di testo subordinati a una clausola REDEFINES, create un nuovo file sorgente che contenga i layout dei record. Il nuovo programma non ha bisogno di descrivere il file utilizzando le istruzioni SELECT e FD. Il programma può semplicem ente contenere le descrizio ni dei record come 01 livelli all'interno di Working-Storage. **Note** È possibile creare un file sorgente per ogni file di dati o creare un file sorgente principale che descriva tutti i file di dati.	Sviluppatore di app

Operazione	Description	Competenze richieste
Compila il file sorgente.	Compila il file sorgente per creare il dizionario dei dati. Ti consigliamo di compilare il file sorgente utilizzando il set di caratteri EBCDIC. Se vengono utilizzate la direttiva IBMCOMP o le direttive ODOSLIDE, è necessario utilizzare queste direttive anche nel file sorgente. **Note** IBMCOMP influisce sulla memorizza zione dei byte dei campi COMP e ODOSLIDE influisce sul padding sulle strutture OCCURS VARIYING. Se queste direttive sono impostate in modo errato, lo strumento di conversione non leggerà correttamente il record di dati. Ciò si traduce in dati errati nel file convertito.	Sviluppatore di app

(Opzione A) Crea il file di struttura utilizzando Classic Data File Tools

Operazione	Description	Competenze richieste
Avvia lo strumento e carica il dizionario.	1. Scegliete l'icona del menu Start di Windows, cercate e scegliete Micro Focus Enterprise Developer, quindi scegliete Classic Data File Tools. 2. Scegliete File, quindi scegliete Record Layout. 3. Nella finestra di dialogo Seleziona un file da cui costruire i layout, in Nome file, selezionate il file IDY (`.idy`) che è stato creato quando avete compilato il file sorgente in precedenz a. Scegliere quindi Open (Apri). 4. Per confermare che Classic Data File Tools utilizza EBCDIC, nella finestra di dialogo Data File Tools, scegliete SÌ se il file IDY è impostato su EBCDIC e Datatools è impostato su ANSI.	Sviluppatore di app
Create il layout di record predefinito.	Utilizza il layout di record predefinito per tutti i record che non corrispondono a nessun layout condizionale. 1. Nella finestra Layout, espandi la struttura dei dati,	Sviluppatore di app

Operazione	Description	Competenze richieste
	quindi individua il livello 01 utilizzato per il layout predefinito. 2. Fate clic con il pulsante destro del mouse sull'elem ento 01, quindi scegliete Nuovo layout. 3. Nella finestra di dialogo New Record Layout Wizard, scegliete Layout predefinito, quindi scegliete Avanti. 4. Scegli Fine. Il layout predefinito viene visualizzato nel riquadro Layout e può essere identific ato dall'icona rossa della cartella.	

Operazione	Description	Competenze richieste
Crea un layout di registrazione condizionale.	Usa il layout di registrazione condizionale quando c'è più di un layout di record in un file. 1. Nel riquadro Layout, espandi la struttura dei dati, quindi individua il livello 01 utilizzato per il layout condizionale. 2. Fate clic con il pulsante destro del mouse sull'elem ento 01, quindi scegliete Nuovo layout. 3. Nella finestra di dialogo New Record Layout Wizard, scegliete Layout condizion ale, quindi scegliete Avanti. 4. Scegli Fine. Il layout condizionale viene visualizz ato nel riquadro Layout e può essere identific ato dall'icona gialla della cartella. 5. Espandi il layout condizion ale, fai clic con il pulsante destro del mouse sul campo in cui devi inserire una condizione, quindi scegli Proprietà. 6. Nella finestra di dialogo Proprietà del campo, inserite la condizione. Verificate che il set di caratteri sia impostato su	Sviluppatore di app

Operazione	Description	Competenze richieste
	EBCDIC, quindi scegliete OK. Accanto al campo con una condizione impostata viene visualizzato un segno di spunta. 7. Ripeti i passaggi da 5 a 6 per tutti gli altri campi che richiedono condizioni per questo layout. 8. Ripetere i passaggi da 1 a 6 per tutti gli altri layout condizionali da aggiungere. 9. Scegliete File, scegliete Salva con nome, quindi salvate il file di struttura su disco.	

(Opzione B) Create il file di struttura utilizzando Data File Tools

Operazione	Description	Competenze richieste
Avvia lo strumento e carica il dizionario.	1. Scegliete l'icona del menu Start di Windows, cercate e scegliete Micro Focus Enterprise Developer, quindi scegliete Data File Tools. 2. Scegliete File, Nuovo, File di struttura. 3. Nella finestra di dialogo Apri, in Nome file, seleziona te il file IDY (`.idy`) creato quando avete compilato il	Sviluppatore di app

Operazione	Description	Competenze richieste
	file sorgente in precedenz a. Scegliere quindi Open (Apri). 4. Per confermare che Data File Tools utilizzi EBCDIC, verifica che il menu a discesa nella sezione Debug File sia impostato su EBCDIC.	
Crea il layout di record predefinito.	Utilizza il layout di record predefinito per tutti i record che non corrispondono a nessun layout condizionale. 1. Nella sezione Layout disponibili nel riquadro sinistro, espandi la struttura dei dati, quindi individua il livello 01 utilizzato per il layout predefinito. 2. Fate clic con il pulsante destro del mouse sull'elem ento 01, quindi scegliete Crea layout predefinito. Il layout predefinito viene visualizzato nel riquadro Layout e può essere identific ato dall'icona blu «D».	Sviluppatore di app

Operazione	Description	Competenze richieste
Crea un layout di registrazione condizionale.	Usa il layout di registrazione condizionale quando c'è più di un layout di record in un file. 1. Nella sezione Layout selezionati nel riquadro di destra, espandi la struttura dei dati, quindi individua il livello 01 utilizzato per il layout condizionale. 2. Fai clic con il pulsante destro del mouse sull'elem ento 01, quindi scegli Crea layout condizionale. Il layout condizionale viene visualizz ato nel riquadro Layout sul lato destro e può essere identificato dall'icona verde «C». 3. Espandi il layout condizion ale, fai clic con il pulsante destro del mouse sul campo in cui devi inserire una condizione, quindi scegli Proprietà. 4. Nella finestra di dialogo Proprietà del campo, inserite la condizione. Verificate che il set di caratteri sia impostato su EBCDIC, quindi scegliete OK. Accanto al campo con una condizione impostata	Sviluppatore di app

Operazione	Description	Competenze richieste
	viene visualizzata un'icona rossa «IF». 5. Ripeti i passaggi da 3 a 4 per tutti gli altri campi che richiedono condizioni per questo layout. 6. Ripetere i passaggi da 1 a 4 per tutti gli altri layout condizionali da aggiungere. 7. Scegliete File, scegliete Salva con nome, quindi salvate il file di struttura su disco.	

(Opzione A) Testate il file di struttura utilizzando Classic Data File Tools

Operazione	Description	Competenze richieste
Prova un file di dati EBCDIC.	Conferma di poter utilizzare il file di struttura per visualizzare correttamente un file di dati di test EBCDIC. 1. Scegliete l'icona del menu Start di Windows, individua te e scegliete Micro Focus Enterprise Developer, quindi scegliete Classic Data Tools. 2. Scegliete File, quindi scegliete Apri. 3. Nella finestra di dialogo Apri, in Nome file, seleziona	Sviluppatore di app

Operazione	Description	Competenze richieste
	te il set di dati EBCDIC, quindi scegliete Apri. 4. Scegliete File, Editor di file di dati, Carica layout di registrazione. 5. Nella finestra di dialogo Apri, in Nome file, seleziona te il file di struttura, quindi scegliete Apri. 6. Per confermare che la modalità set di caratteri sia impostata su EBCDIC, verificate che il menu a discesa sia impostato su EBCDIC. Puoi vedere i dati grezzi dei record nel riquadro a sinistra e i dati formattati nel riquadro a destra. 7. Scegli vari record per assicurarti che tutti i formati siano renderizzati con il layout corretto.	

(Opzione B) Testate il file di struttura utilizzando Data File Tools

Operazione	Description	Competenze richieste
Prova un file di dati EBCDIC.	Conferma di poter utilizzare il file di struttura per visualizzare correttamente un file di dati di test EBCDIC.	Sviluppatore di app

Operazione	Description	Competenze richieste
	1. Scegliete l'icona del menu Start di Windows, individua te e selezionate Micro Focus Enterprise Developer , quindi scegliete Data File Tools. 2. Scegliete File, Apri, File di dati. 3. Nella finestra di dialogo Apri file di dati, nella scheda Locale, per Nome file, scegliete Sfoglia per trovare la posizione del file di test EBCDIC. 4. Per Structure File (opzional e), scegliete Sfoglia per trovare la posizione del file di struttura. 5. Nella sezione Dettagli del file, inserisci i dettagli del file e conferma che la codifica sia impostata su EBCDIC. 6. Scegliete la modalità Open Shared o Open Exclusive a seconda delle vostre esigenze. 7. Verifica che il menu a discesa nella sezione Aspetto della barra degli strumenti sia impostato su EBCDIC. Vedrai i dati grezzi dei record nel riquadro a sinistra e i dati	

Operazione	Description	Competenze richieste
	formattati nel riquadro a destra. 8. Scegli vari record per assicurarti che tutti i formati siano renderizzati con il layout corretto.	

Prova la conversione dei file di dati

Operazione	Description	Competenze richieste
Verifica la conversione di un file EBCDIC.	1. Scegliete l'icona del menu Start di Windows, individua te e selezionate Micro Focus Enterprise Developer , quindi scegliete Classic Data Tools. 2. Scegliete Strumenti, quindi scegliete Converti. 3. Nella finestra di dialogo Conversione file di dati, nella sezione File di input, per Nome file, scegliete Sfoglia per trovare e selezionare il file di input EBCDIC. Verificate che il set di caratteri sia impostato su EBCDIC. 4. Nella sezione Conversio ne del set di caratteri , selezionate le caselle di controllo Converti set di caratteri e i record	Sviluppatore di app

Operazione	Description	Competenze richieste
	contengono elementi di dati non testuali. Scegliete Seleziona il layout per la conversione, quindi scegliete Sfoglia per trovare e selezionare il file di struttura. 5. Nella sezione Nuovo file, in Nome file, inserisci il percorso e il nome del file di output ASCII che desideri creare. Per impostazione predefinita, lo strumento di conversione utilizza lo stesso formato del file di input. Per il test, lasciate le opzioni impostate sui valori predefiniti. 6. Scegli Converti. 7. Segui i passaggi della sezione (Opzione A) Verifica del file di struttura utilizzando Classic Data File Tools o (Opzione B) Verifica il file di struttura utilizzan do gli strumenti per i file di dati, ma carica il file di output ASCII anziché il file EBCDIC. 8. Carica i file EBCDIC e ASCII nell'Editor dei file di dati, quindi confronta i file uno accanto all'altro	

Operazione	Description	Competenze richieste
	per verificare l'accuratezza della conversione.	

Risorse correlate

- Micro Focus (documentazione Micro Focus)
- Mainframe e codice legacy (post sul blog AWS)
- AWS Prescriptive Guidance (documentazione AWS)
- Documentazione AWS (documentazione AWS)
- Riferimento generale AWS (documentazione AWS)
- Glossario AWS (documentazione AWS)

Implementa un ambiente per applicazioni Blu Age containerizzate utilizzando Terraform

Richard Milner-Watts, Amazon Web Services

Riepilogo

La migrazione dei carichi di lavoro mainframe legacy in architetture cloud moderne può eliminare i costi di manutenzione di un mainframe, costi che aumentano solo con l'invecchiamento dell'ambiente. Tuttavia, la migrazione dei lavori da un mainframe può porre sfide uniche. Le risorse interne potrebbero non conoscere la logica del lavoro e le elevate prestazioni dei mainframe in queste attività specializzate possono essere difficili da replicare rispetto a quelle di uso generalizzato. CPUs Riscrivere questi lavori può essere un'impresa ardua e richiedere un notevole impegno.

Blu Age converte i carichi di lavoro mainframe legacy in codice Java moderno, che puoi quindi eseguire come contenitore.

Questo modello fornisce un esempio di architettura serverless per l'esecuzione di un'applicazione containerizzata che è stata modernizzata con lo strumento Blu Age. I file HashiCorp Terraform inclusi creeranno un'architettura sicura per l'orchestrazione dei contenitori Blu Age, supportando sia attività in batch che servizi in tempo reale.

Per ulteriori informazioni sulla modernizzazione dei carichi di lavoro utilizzando Blu Age e i servizi AWS, consulta queste pubblicazioni AWS Prescriptive Guidance:

- Esecuzione di carichi di lavoro mainframe modernizzati con Blu Age sull'infrastruttura serverless AWS
- Containerizza i carichi di lavoro mainframe che sono stati modernizzati da Blu Age

Per ricevere assistenza sull'utilizzo di Blu Age per modernizzare i carichi di lavoro mainframe, contatta il team Blu Age selezionando Contatta i nostri esperti sul sito Web Blu Age. Per assistenza sulla migrazione dei carichi di lavoro modernizzati su AWS, l'integrazione con i servizi AWS e il loro trasferimento in produzione, contatta il tuo account manager AWS o compila il modulo AWS Professional Services.

Prerequisiti e limitazioni

Prerequisiti

- L'applicazione di esempio di Blu Age containerizzata fornita dai carichi di lavoro mainframe Containerize che sono stati modernizzati dal modello Blu Age. L'applicazione di esempio fornisce la logica per gestire l'elaborazione di input e output per l'applicazione modernizzata e può integrarsi con questa architettura.
- Terraform è necessario per distribuire queste risorse.

Limitazioni

- Amazon Elastic Container Service (Amazon ECS) impone dei limiti alle risorse delle attività che possono essere rese disponibili per il container. Queste risorse includono CPU, RAM e storage. Ad esempio, quando si utilizza Amazon ECS con AWS Fargate, si applicano i limiti delle risorse delle attività.

Versioni del prodotto

Questa soluzione è stata testata con le seguenti versioni:

- Terraform 1.3.6
- Terraform AWS Provider 4.46.0

Architecture

Stack tecnologico di origine

- Età blu
- Terraform

Stack tecnologico Target

- Amazon Aurora PostgreSQL-Compatible Edition
- AWS Backup
- Amazon Elastic Container Registry (Amazon ECR)
- Amazon ECS
- AWS Identity and Access Management Service (IAM)
- Server di gestione delle chiavi AWS (AWS KMS)

- AWS Secrets Manager
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Storage Service (Amazon S3)
- AWS Step Functions
- AWS Systems Manager

Architettura Target

Il diagramma seguente mostra l'architettura della soluzione.

1. La soluzione implementa i seguenti ruoli IAM:
   - Ruolo dell'attività Batch
   - Ruolo di esecuzione delle attività Batch
   - Ruolo dell'attività di servizio
   - Ruolo di esecuzione dell'attività di servizio
   - Ruolo Step Functions
   - Ruolo di AWS Backup
   - Ruolo RDS Enhanced Monitoring.

   I ruoli sono conformi ai principi di accesso con privilegi minimi.
2. Amazon ECR viene utilizzato per archiviare l'immagine del contenitore orchestrata da questo modello.
3. AWS Systems Manager Parameter Store fornisce dati di configurazione su ciascun ambiente alla definizione delle attività di Amazon ECS in fase di esecuzione.
4. AWS Secrets Manager fornisce dati di configurazione sensibili sull'ambiente alla definizione delle attività di Amazon ECS in fase di esecuzione. I dati sono stati crittografati da AWS KMS.
5. I moduli Terraform creano definizioni di attività Amazon ECS per tutte le attività in tempo reale e in batch.
6. Amazon ECS esegue un'attività in batch utilizzando AWS Fargate come motore di elaborazione. Si tratta di un'attività di breve durata, avviata come richiesto da AWS Step Functions.
7. Amazon Aurora, compatibile con PostgreSQL, fornisce un database per supportare l'applicazione modernizzata. Questo sostituisce i database mainframe come IBM Db2 o IBM IMS DB.

8. Amazon ECS offre un servizio di lunga durata per fornire un carico di lavoro modernizzato in tempo reale. Queste applicazioni stateless vengono eseguite in modo permanente con contenitori distribuiti tra le zone di disponibilità.
9. Un Network Load Balancer viene utilizzato per concedere l'accesso al carico di lavoro in tempo reale. Il Network Load Balancer supporta protocolli precedenti, come IBM CICS. In alternativa, puoi utilizzare un Application Load Balancer con carichi di lavoro basati su HTTP.
10. Amazon S3 fornisce lo storage di oggetti per gli input e gli output dei processi. Il contenitore dovrebbe gestire le operazioni pull and push in Amazon S3 per preparare la directory di lavoro per l'applicazione Blu Age.
11. Il servizio AWS Step Functions viene utilizzato per orchestrare l'esecuzione delle attività di Amazon ECS per elaborare carichi di lavoro in batch.
12. Gli argomenti SNS per ogni carico di lavoro batch vengono utilizzati per integrare l'applicazione modernizzata con altri sistemi, come la posta elettronica, o per avviare azioni aggiuntive, come la consegna di oggetti di output da Amazon S3 a FTP.

> **Note**
>
> Per impostazione predefinita, la soluzione non ha accesso a Internet. Questo modello presuppone che il cloud privato virtuale (VPC) sia connesso ad altre reti utilizzando un servizio come AWS Transit Gateway. Pertanto, vengono implementati endpoint VPC a più interfacce per garantire l'accesso ai servizi AWS utilizzati dalla soluzione. Per attivare l'accesso diretto a Internet, puoi utilizzare l'interruttore nel modulo Terraform per sostituire gli endpoint VPC con un gateway Internet e le risorse associate.

Automazione e scalabilità

L'uso di risorse serverless in questo modello contribuisce a garantire che, grazie alla scalabilità orizzontale, vi siano pochi limiti alla scalabilità di questo progetto. In questo modo si riducono i fastidiosi problemi dei vicini, come la concorrenza per le risorse di elaborazione che si potrebbe riscontrare sul mainframe originale. Le attività Batch possono essere pianificate per l'esecuzione simultanea in base alle esigenze.

I singoli contenitori sono limitati dalle dimensioni massime supportate da Fargate. Per ulteriori informazioni, consulta la sezione Task CPU e memoria nella documentazione di Amazon ECS.

Per scalare orizzontalmente i carichi di lavoro in tempo reale, puoi aggiungere contenitori.

Tools (Strumenti)

Servizi AWS

- Amazon Aurora PostgreSQL Compatible Edition è un motore di database relazionale completamente gestito e conforme ad ACID che ti aiuta a configurare, gestire e scalare le distribuzioni PostgreSQL.
- AWS Backup è un servizio completamente gestito che ti aiuta a centralizzare e automatizzare la protezione dei dati su Servizi AWS, nel cloud e in locale.
- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- Amazon Elastic Container Service (Amazon ECS) è un servizio rapido e scalabile di gestione dei container che ti aiuta a eseguire, arrestare e gestire container in un cluster.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- AWS Secrets Manager ti aiuta a sostituire le credenziali codificate nel codice, comprese le password, con una chiamata API a Secrets Manager per recuperare il segreto a livello di codice.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Step Functions è un servizio di orchestrazione senza server che ti aiuta a combinare AWS Lambda funzioni e altro Servizi AWS per creare applicazioni aziendali critiche.
- AWS Systems Manager Parameter Store fornisce uno storage sicuro e gerarchico per la gestione dei dati di configurazione e la gestione dei segreti.

Altri servizi

- HashiCorp Terraform è uno strumento Infrastructure as Code (IaC) che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud. Questo modello utilizza Terraform per creare l'architettura di esempio.

Archivio di codice

Il codice sorgente di questo pattern è disponibile nel repository GitHub Blu Age Sample ECS Infrastructure (Terraform).

Best practice

- Per gli ambienti di test, utilizza funzionalità come la `forceDate` possibilità di configurare l'applicazione modernizzata per generare risultati di test coerenti eseguendo sempre per un periodo di tempo noto.
- Ottimizza ogni attività individualmente per consumare la quantità ottimale di risorse. Puoi utilizzare Amazon CloudWatch Container Insights per ottenere indicazioni su potenziali colli di bottiglia.

Epiche

Prepara l'ambiente per l'implementazione

Operazione	Description	Competenze richieste
Clona il codice sorgente della soluzione.	Clona il codice della soluzione dal GitHub progetto.	DevOps ingegnere
Avvia l'ambiente distribuendo risorse per archiviare lo stato di Terraform.	1. Apri una finestra di terminale e conferma che Terraform sia installato e che AWS le credenziali siano disponibili. 2. Accedi alla cartella `bootstrap-terraform` . 3. Modifica il file `main.tf` se desideri cambiare i nomi del bucket S3 (`<accountId>-terraform-backend`) e della tabella Amazon DynamoDB (). `terraform-lock` 4. Esegui il `terraform apply` comando per	DevOps ingegnere

Operazione	Description	Competenze richieste
	distribuire le risorse. Prendi nota del bucket S3 e dei nomi delle tabelle DynamoDB.	

Implementa l'infrastruttura della soluzione

Operazione	Description	Competenze richieste
Rivedi e aggiorna la configura zione di Terraform.	Nella directory principale, apri il file, `main.tf`, esamina il contenuto e valuta la possibili tà di apportare i seguenti aggiornamenti: 1. Aggiorna il Regione AWS file cercando e sostituendo la stringa `eu-west-1` con la regione desiderata che desideri utilizzare. 2. Aggiorna il nome del bucket nel `Terraform Backend` blocco se il valore predefinito è stato modificat o nell'epopea precedente. 3. Aggiorna il `dynamodb_ table` valore se il valore predefinito è stato modificat o nell'epopea precedente. 4. Aggiorna il valore della `stack_prefix` variabile nella stringa che desideri. Questa stringa verrà aggiunta ai nomi di tutte le	DevOps ingegnere

Operazione	Description	Competenze richieste
	risorse create da questo modello. 5. Aggiorna il valore di `vpc_cidr` Questo dovrebbe essere almeno un intervallo di `/24` indirizzi. 6. Rivedi la `Locals` sezione. Viene utilizzato per definire le attività di Blu Age che verranno implementate. La soluzione eseguirà un'iterazione sull'ogge tto elenco`bluage_ba tch_modules` , creando le risorse associate (macchina a stati Step Functions, definizione dell'attività e argomento SNS) per ogni elemento dell'elenco. In alcuni casi, potresti voler regolare le variabili per ambienti diversi. Ad esempio, per forzare il runtime negli ambienti di test, puoi modificare il valore della `force_execution_ti me` variabile. 7. Per attivare l'accesso a Internet, modificate il valore `direct_in ternet_access_requ ired` da `false` a`true`. Questo implementerà un	

Operazione	Description	Competenze richieste
	gateway Internet, insieme ai gateway NAT e alle tabelle di routing che attivano l'accesso pubblico a Internet per l'infrastruttura. Per impostazione predefini ta, la soluzione implement erà gli endpoint VPC di interfaccia in un VPC senza accesso diretto a Internet. 8. Per concedere l'accesso a qualsiasi carico di lavoro client-server servito tramite Elastic Load Balancing, aggiorna i valori `additional_nlb_igr ess_cidrs` di con le reti CIDR che dovrebbero essere consentiti.	
Distribuisci il file Terraform.	Dal tuo terminale, esegui il `terraform apply` comando per distribuire tutte le risorse. Esamina le modifiche generate da Terraform e inserisci yes per avviare la build. Tieni presente che possono essere necessari più di 15 minuti per implementare questa infrastruttura.	DevOps ingegnere

(Facoltativo) Implementa un'applicazione containerizzata Blu Age valida

Operazione	Description	Competenze richieste
Invia l'immagine del contenitore Blu Age ad Amazon ECR.	Inserisci il contenitore nel repository Amazon ECR che hai creato nell'epopea precedente. Per istruzioni, consulta la documentazione di Amazon ECR. Prendi nota dell'URI dell'immagine del contenitore.	DevOps ingegnere
Aggiorna Terraform in modo che faccia riferimento all'immagine del contenitore Blu Age.	Aggiorna il file `main.tf` in modo che faccia riferimento all'immagine del contenitore che hai caricato.	DevOps ingegnere
Ridistribuisci il file Terraform.	Dal tuo terminale, esegui `terraform apply` per distribuire tutte le risorse. Esamina gli aggiornamenti suggeriti da Terraform, quindi inserisci yes per procedere con la distribuzione.	DevOps ingegnere

Risorse correlate

- Età blu
- Esecuzione di carichi di lavoro mainframe modernizzati con Blu Age sull'infrastruttura serverless AWS
- Containerizza i carichi di lavoro mainframe che sono stati modernizzati da Blu Age

Genera approfondimenti sui z/OS dati Db2 utilizzando Modernizzazione del mainframe AWS Amazon Q in Quick Sight

Shubham Roy, Roshna Razack e Santosh Kumar Singh, Amazon Web Services

Riepilogo

Nota: il Modernizzazione del mainframe AWS servizio (esperienza Managed Runtime Environment) non è più aperto a nuovi clienti. Per funzionalità simili a Modernizzazione del mainframe AWS Service (Managed Runtime Environment experience), esplora Modernizzazione del mainframe AWS Service (Self-Managed Experience). I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta Modifica della Modernizzazione del mainframe AWS disponibilità.

Se la tua organizzazione ospita dati aziendali critici in un ambiente mainframe IBM Db2, ottenere informazioni approfondite da tali dati è fondamentale per promuovere la crescita e l'innovazione. Sbloccando i dati del mainframe, puoi creare business intelligence più veloce, sicura e scalabile per accelerare il processo decisionale, la crescita e l'innovazione basati sui dati nel cloud Amazon Web Services ().AWS

Questo modello presenta una soluzione per generare approfondimenti aziendali e creare narrazioni condivisibili a partire dai dati mainframe in IBM Db2 for tables. z/OS Le modifiche ai dati del mainframe vengono trasmesse in streaming all'argomento Amazon Managed Streaming for Apache Kafka (Amazon Modernizzazione del mainframe AWS MSK) utilizzando Data Replication with Precisly. Utilizzando lo streaming ingestion di Amazon Redshift, i dati tematici di Amazon MSK vengono archiviati in tabelle di data warehouse Serverless di Amazon Redshift per l'analisi in Amazon Quick Sight.

Dopo che i dati sono disponibili in Quick Sight, puoi utilizzare istruzioni in linguaggio naturale con Amazon Q in Quick Sight per creare riepiloghi dei dati, porre domande e generare storie di dati. Non è necessario scrivere query SQL o imparare a usare uno strumento di business intelligence (BI).

Contesto aziendale

Questo modello presenta una soluzione per i casi d'uso di analisi dei dati mainframe e analisi dei dati. Utilizzando il modello, crei una dashboard visiva per i dati della tua azienda. Per dimostrare la soluzione, questo modello utilizza un'azienda sanitaria che fornisce piani medici, dentistici e oculistici ai suoi membri negli Stati Uniti. In questo esempio, i dati demografici dei membri e le informazioni sul piano vengono archiviati in IBM Db2 for data tables. z/OS La dashboard visiva mostra quanto segue:

- Distribuzione dei membri per regione
- Distribuzione dei membri per genere
- Distribuzione dei membri per età
- Distribuzione dei membri per tipo di piano
- Membri che non hanno completato l'immunizzazione preventiva

Per esempi di distribuzione dei membri per regione e membri che non hanno completato l'immunizzazione preventiva, vedere la sezione Informazioni aggiuntive.

Dopo aver creato la dashboard, si genera una storia basata sui dati che spiega le informazioni ricavate dall'analisi precedente. La storia dei dati fornisce raccomandazioni per aumentare il numero di membri che hanno completato le vaccinazioni preventive.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS. Questa soluzione è stata creata e testata su Amazon Linux 2 su Amazon Elastic Compute Cloud (Amazon EC2).
- Un cloud privato virtuale (VPC) con una sottorete accessibile dal sistema mainframe.
- Un database mainframe con dati aziendali. Per i dati di esempio utilizzati per creare e testare questa soluzione, consulta la sezione Allegati.
- Change Data Capture (CDC) abilitato nelle tabelle z/OS Db2. Per abilitare CDC su Db2 z/OS, consulta la documentazione IBM.
- Precisly Connect CDC per l' z/OS installazione sul z/OS sistema che ospita i database di origine. Il Precisly Connect CDC for z/OS image viene fornito come file zip all'interno di Modernizzazione del mainframe AWS - Data Replication for IBM z/OS Amazon Machine Image (AMI). Per installare Precisly Connect CDC for z/OS sul mainframe, consulta la documentazione di installazione di Precisly Connect.

Limitazioni

- I dati Db2 del mainframe devono essere in un tipo di dati supportato da Precisly Connect CDC. Per un elenco dei tipi di dati supportati, consulta la documentazione di Precisly Connect CDC.
- I tuoi dati su Amazon MSK devono essere in un tipo di dati supportato da Amazon Redshift. Per un elenco dei tipi di dati supportati, consulta la documentazione di Amazon Redshift.

- Amazon Redshift ha comportamenti e limiti di dimensione diversi per diversi tipi di dati. Per ulteriori informazioni, consulta la documentazione di Amazon Redshift.
- I dati quasi in tempo reale in Quick Sight dipendono dall'intervallo di aggiornamento impostato per il database Amazon Redshift.
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità per regione, vedi Servizi AWS per regione. Amazon Q in Quick Sight non è attualmente disponibile in tutte le regioni che supportano Quick Sight. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- Modernizzazione del mainframe AWS Replica dei dati con Precisly versione 4.1.44
- Python versione 3.6 o successiva
- Apache Kafka versione 3.5.1

Architecture

Architettura Target

Il diagramma seguente mostra un'architettura per generare informazioni aziendali dai dati mainframe utilizzando Modernizzazione del mainframe AWS Data Replication with Precisly e Amazon Q in Quick Sight.

Il diagramma mostra il flusso di lavoro seguente:

1. L'agente Precisly Log Reader legge i dati dai log Db2 e li scrive in uno storage transitorio su un file system OMVS sul mainframe.
2. Il Publisher Agent legge i log Db2 non elaborati dallo storage transitorio.
3. Il daemon del controller locale autentica, autorizza, monitora e gestisce le operazioni.
4. L'agente Apply viene distribuito su Amazon EC2 utilizzando l'AMI preconfigurata. Si connette con Publisher Agent tramite il daemon del controller utilizzando TCP/IP. L'agente Apply invia i dati ad Amazon MSK utilizzando più worker per un throughput elevato.
5. I lavoratori scrivono i dati nell'argomento Amazon MSK in formato JSON. Come destinazione intermedia per i messaggi replicati, Amazon MSK offre funzionalità di failover automatizzate e ad alta disponibilità.

6. L'ingestione di streaming di Amazon Redshift fornisce l'inserimento di dati a bassa latenza e alta velocità da Amazon MSK a un database Amazon Redshift Serverless. Una procedura memorizzata in Amazon Redshift esegue la riconciliazione dei dati di modifica del mainframe (insert/update/deletes) nelle tabelle Amazon Redshift. Queste tabelle Amazon Redshift fungono da fonte di analisi dei dati per Quick Sight.
7. Gli utenti accedono ai dati in Quick Sight per analisi e approfondimenti. Puoi utilizzare Amazon Q in Quick Sight per interagire con i dati utilizzando istruzioni in linguaggio naturale.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e scalarli rapidamente verso l'esterno o verso l'esterno.
- AWS Key Management Service (AWS KMS) consente di creare e controllare chiavi crittografiche per proteggere i dati.
- Amazon Managed Streaming for Apache Kafka (Amazon MSK) è un servizio completamente gestito che ti aiuta a creare ed eseguire applicazioni che utilizzano Apache Kafka per elaborare dati di streaming.
- Amazon Quick Sight è un servizio di business intelligence (BI) su scala cloud che ti aiuta a visualizzare, analizzare e generare report sui dati in un'unica dashboard. Questo modello utilizza le funzionalità di BI generativa di Amazon Q in Quick Sight.
- Amazon Redshift Serverless è un'opzione serverless di Amazon Redshift che rende più efficiente l'esecuzione e la scalabilità delle analisi in pochi secondi senza la necessità di configurare e gestire l'infrastruttura di data warehouse.
- Con Gestione dei segreti AWS puoi sostituire le credenziali nel codice,incluse le password, con una chiamata API a Secrets Manager in modo da recuperare il segreto a livello di codice.

Altri strumenti

- Precisly Connect CDC raccoglie e integra i dati dai sistemi legacy in piattaforme cloud e dati.

Deposito di codici

Il codice per questo pattern è disponibile nel repository GitHub DataInsightsMainframe_ _change_data_reconciliation. Il codice è una procedura memorizzata in Amazon Redshift. Questa procedura memorizzata riconcilia le modifiche ai dati del mainframe (inserimenti, aggiornamenti ed eliminazioni) da Amazon MSK alle tabelle Amazon Redshift. Queste tabelle Amazon Redshift fungono da fonte di analisi dei dati per Quick Sight.

Best practice

- Segui le best practice durante la configurazione del cluster Amazon MSK.
- Segui le best practice di analisi dei dati di Amazon Redshift per migliorare le prestazioni.
- Quando crei i ruoli AWS Identity and Access Management (IAM) per la configurazione Precisly, segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.

Epiche

Configura la replica Modernizzazione del mainframe AWS dei dati con Precisly on Amazon EC2

Operazione	Description	Competenze richieste

Configura un gruppo di sicurezza.	Per connetterti al daemon del controller e al cluster Amazon MSK, crea un gruppo di sicurezza per l'istanza. EC2 Aggiungi le seguenti regole in entrata e in uscita:
	• Regola 1 in entrata:
	  • Per Tipo, scegli TCP personalizzato.
	  • Per Protocol (Protocollo), selezionare TCP.
	  • Per Port range, scegliete 2626 (porta predefinita per il daemon Precisly controller) o il numero
		DevOps ingegnere, AWS DevOps

Operazione	Description	Competenze richieste
	di porta del demone del controller in esecuzione sul mainframe. • Per Source, scegli il blocco CIDR. • Regola 2 in entrata: • Per Type (Tipo), scegliere Custom TCP (TCP personalizzato). • Per Protocollo, scegli SSH. • Per Port range, scegli 22. • Per Sorgente, scegli l'indirizzo IP o l'elenco dei prefissi. • Regola 3 in entrata: • Per Type (Tipo), scegliere Custom TCP (TCP personalizzato). • Per Protocol (Protocollo), selezionare TCP. • Per Port range, scegli 9092-9098. • Per Source, scegli il blocco CIDR. • Regola 1 in uscita: • Per Type (Tipo), scegliere Custom TCP (TCP personalizzato). • Per Protocol (Protocollo), selezionare TCP.	

Operazione	Description	Competenze richieste
	• Per Port range, scegli 9092-9098. • Per Source, scegli il blocco CIDR. • Regola 2 in uscita: • Per Type (Tipo), scegliere Custom TCP (TCP personalizzato). • Per Protocol (Protocollo), selezionare TCP. • Per Port range, scegliete 2626 (porta predefinita per il demone Precisly controller) o il numero di porta del demone del controller in esecuzione sul mainframe. • Per Source, scegli il blocco CIDR.	
	Annota il nome del gruppo di sicurezza. Dovrai fare riferimento al nome quando avvii l' EC2 istanza e configuri il cluster Amazon MSK.	

Operazione	Description	Competenze richieste
Crea una policy IAM e un ruolo IAM.	1. Per creare una policy IAM e un ruolo IAM, segui le istruzioni nella AWS documentazione. La policy IAM concede l'accesso per creare argomenti sul cluster Amazon MSK e inviare dati a tali argomenti. 2. Dopo aver creato il ruolo IAM, associa la policy ad esso. Nota il nome del ruolo IAM. Questo ruolo verrà utilizzato come profilo dell'istanza IAM all'avvio dell' EC2 istanza.	DevOps ingegnere, amministratore di sistema AWS

Operazione	Description	Competenze richieste
Fornisci un' EC2 istanza.	Per fornire un' EC2 istanza per eseguire Precisly CDC e connettersi ad Amazon MSK, procedi come segue: 1. Accedi e iscriviti a Marketplace AWSModernizzazione del mainframe AWS – Data Replication for IBM z/OS. 2. Seleziona l'AMI dagli abbonamenti gestiti e scegli Launch new instance. 3. Fornisci altri dettagli di configurazione, come il nome dell'istanza, il tipo di istanza, la coppia di chiavi, il VPC e le sottoreti . Per ulteriori informazioni, consulta la EC2 documentazione di Amazon. 4. Nell'elenco a discesa, scegli il gruppo di sicurezza che hai creato in precedenza. 5. In Dettagli avanzati, profilo dell'istanza IAM, devi selezionare il ruolo che hai creato in precedenza. 6. Scegliere Launch Instance (Avvia istanza).	Amministratore, DevOps ingegnere di AWS

Configurare Amazon MSK

Operazione	Description	Competenze richieste
Crea il cluster Amazon MSK.	Per creare un cluster Amazon MSK, procedi come segue: 1. Accedi a e accedi ad Amazon MSK. Console di gestione AWS 2. Scegli Crea cluster. 3. Per il metodo di creazione del cluster, scegli Creazione personalizzata e, per il tipo di cluster, scegli Provision ed. 4. Fornire un nome per il cluster. 5. Aggiorna le impostazioni del cluster secondo necessità e mantieni i valori predefiniti per le altre impostazioni. 6. Nota la < versione di Kafka>. Ti servirà durante la configurazione del client Kafka. 7. Scegli Next (Successivo). 8. Scegli lo stesso VPC e le stesse sottoreti che hai usato per l' EC2 istanza Precisly e scegli il gruppo di sicurezza che hai creato in precedenza. 9. Nella sezione Impostazioni di sicurezza, abilita	AWS DevOps, amministratore del cloud

Operazione	Description	Competenze richieste
	l'autenticazione basata sui ruoli SASL/SCRAM e IAM. Precisly Connect CDC utilizza SASL/SCRAM (Simple Authentication and Security Layer/ Salted Challenge Response Mechanism) e IAM è necessario per la connessione ad Amazon Redshift. 10. Scegli Next (Successivo). 11. Per la revisione, scegli il metodo di consegna dei log di Monitoring and Broker. 12. Scegli Avanti, quindi scegli Crea il cluster. La creazione di un cluster con provisioning tipico richiede fino a 15 minuti. Dopo la creazione del cluster, lo stato del cluster cambia da Creato ad Attivo.	

Operazione	Description	Competenze richieste
Configura SASL/SCRAM l'autenticazione.	Per configurare SASL/SCRAM l'autenticazione per un cluster Amazon MSK, procedi come segue: 1. Per impostare un segreto in Secrets Manager, segui le istruzioni nella AWS documentazione. 2. Apri la console Amazon MSK e seleziona il cluster Amazon MSK creato in precedenza. 3. Scegliere la scheda Properties (Proprietà). 4. Scegli i segreti associati, scegli i segreti, seleziona la chiave segreta che hai creato, quindi scegli Segreti associati. Verrà visualizzato un messaggio di successo simile al seguente: `Successfully associated 1 secret for cluster <chosen cluster name>` 5. Scegli il nome del cluster. 6. Nel riepilogo del cluster, scegli Visualizza informazioni sul client.	Architetto del cloud

Operazione	Description	Competenze richieste
	7. Nota la stringa di connessione all'endpoint privato per il tipo di autenticazione SASL/SCRAM.	

Operazione	Description	Competenze richieste

Crea l'argomento Amazon MSK.	Per creare l'argomento Amazon MSK, procedi come segue:
	1. Connettiti all' EC2 istanza creata in precedenza e installa gli aggiornamenti più recenti eseguendo il comando seguente:
	`sudo yum update -y`
	2. Installa le librerie Java e Kafka eseguendo il seguente comando:
	`sudo yum install -y java-11 librdkafka librdkafka-devel`
	3. Per creare una cartella denominata `kafka` in `/home/ec2-user` , accedete a quella cartella ed eseguite il seguente comando:
	`mkdir kafka;cd kafka`
	4. Scarica la libreria `kafka` client `kafka` nella cartella, sostituendola `<YOUR MSK VERSION>` con la versione di Kafka che hai annotato durante la creazione del cluster Amazon MSK:
		Amministratore cloud

Operazione	Description	Competenze richieste

	`wget https://archive.apache.org/dist/kafka//kafka_2.13-<YOUR MSK VERSION>.tgz`
	5. Per estrarre il file scaricato, esegui il comando seguente, sostituendo: `YOUR MSK VERSION>`
	`tar -xzf kafka_2.13-<YOUR MSK VERSION>.tgz`
	6. Per accedere alla `kafka libs` directory e scaricare il file Java Archive (JAR) di autenticazione Java IAM, esegui i seguenti comandi, sostituendo`<YOUR MSK VERSION>`:
	`cd kafka_2.13-<YOUR MSK VERSION>/libs`
	`wget https://github.com/aws/aws-msk-iam-auth/releases/download/v1.1.1/aws-msk-iam-auth-1.1.1-all.jarkafka`
	7. Per accedere alla `bin` directory Kafka e creare il `client.properties` file, esegui i seguenti comandi:	

Operazione	Description	Competenze richieste

	`cd /home/ec2-user/kafka/kafka_2.13-<YOUR MSK VERSION>/bin`
	`cat >client.properties`
	
	8. Aggiorna il `client.properties` file con i seguenti contenuti:
	
	`security.protocol=SASL_SSL`
	`sasl.mechanism=AWS_MSK_IAM`
	`sasl.jaas.config=software.amazon.msk.auth.iam.IAMLoginModule required;`
	`sasl.client.callback.handler.class=software.amazon.msk.auth.iam.IAMClientCallbackHandler`
	
	9. Per creare un argomento Kafka, accedi al contenitore Kafka ed esegui il comando seguente, sostituendolo `<kafka broker>` con l'endpoint privato del server bootstrap IAM che hai notato durante la creazione del cluster Amazon MSK:
	
	`./kafka-topics.sh --bootstrap-server <kafka broker> --command-config`	

Operazione	Description	Competenze richieste
	`client.properties --create --replication-factor 3 –partitions 6 --topic <topic name>` 10. Quando viene visualizzato il messaggio, annota il nome dell'argomento `Created topic <topic name>`.	

Configura il motore di applicazione precisa su Amazon EC2

Operazione	Description	Competenze richieste
Configura gli script Precisly per replicare le modifiche ai dati.	Per configurare gli script CDC di Precisly Connect per replicare i dati modificati dal mainframe all'argomento Amazon MSK, procedi come segue: 1. Per creare con precision e il nome di una cartella e passare a quella cartella, esegui il seguente comando: `mkdir /home/ec2-user/precisely;cd /home/ec2-user/precisely` 2. Per creare due cartelle all'interno di una cartella chiamata `scripts` con precisione e poi	Sviluppatore di app, architetto cloud

Operazione	Description	Competenze richieste

	passare alla `scripts` cartella, esegui il seguente comando: `ddls`
	
	`mkdir scripts;mkdir ddls;cd scripts`
	
	3. Per creare un file denominato `sqdata_kafka_producer.conf` nella `scripts` cartella, esegui il seguente comando:
	
	`cat >sqdata_kafka_producer.conf`
	
	4. Aggiorna `sqdata_kafka_producer.conf` il file con il seguente contenuto:
	
	`builtin.features=SASL_SCRAM`
	`security.protocol=SASL_SSL`
	`sasl.mechanism=SCRAM-SHA-512`
	`sasl.username=<User Name>`
	`sasl.password=<Password>`
	`metadata.broker.list=<SASL/SCRAM Bootstrap servers>`	

Operazione	Description	Competenze richieste
	Note Aggiorna `<SASL/SCRAM Bootstrap servers>` con l'elenco di SASL/SCRAM broker Amazon MSK che hai configurato in precedenza. Aggiorna `<User Name>` e `<Password>` con il nome utente e la password che hai impostato in precedenza in Secrets Manager. 5. Crea un `script.sqd` file nella `scripts` cartella. `cat >script.sqd` Apply Engine li utilizza `script.sqd` per elaborare i dati di origine e replicare i dati di origine sulla destinazione. Per un esempio di script Apply Engine, consultate la sezione Informazioni aggiuntive.	

Operazione	Description	Competenze richieste

	6. Per passare alla `ddls` cartella e creare un file.ddl per ogni tabella Db2, esegui i seguenti comandi:
	
	`cd /home/ec2-user/precisely/ddls`
	`cat >mem_details.ddl`
	`cat >mem_plans.ddl`
	
	Ad esempio i file.ddl, consulta la sezione Informazioni aggiuntive.	

Operazione	Description	Competenze richieste

Genera la chiave ACL di rete.	Per generare la chiave dell'elenco di controllo degli accessi alla rete (network ACL), procedi come segue:
	
	1. Per esportare il percorso `sqdata` di installaz ione, esegui il comando seguente:
	
	`export PATH=$PAT H:/usr/sbin:/opt/p recisely/di/sqdata/ bin`
	
	2. Per passare alla `/home/ ec2-user` directory e generare la chiave ACL di rete, esegui i seguenti comandi:
	
	`cd /home/ec2-user sqdutil keygen -- force`
	
	Dopo la generazione delle chiavi pubbliche e private, viene visualizzato il seguente messaggio:
	
	`SQDUT04I Generating a private key in file / home/ec2-user/.nacl/ id_nacl SQDC017I sqdutil(p id=27344) terminated successfully`
		Architetto del cloud, AWS DevOps

Operazione	Description	Competenze richieste
	3. Annota la chiave pubblica generata che è memorizza ta nella `.nacl` cartella.	

Prepara l'ambiente di origine del mainframe

Operazione	Description	Competenze richieste
Configurare le impostazioni predefinite nella schermata ISPF.	Per configurare le impostazi oni predefinite nell'Interactive System Productivity Facility (ISPF), segui le istruzioni nella documentazione di Precisly.	Amministratore di sistema mainframe
Configura il demone del controller.	Per configurare il demone del controller, procedi come segue: 1. Nella schermata del menu principale di SQData z/OS, scegliete l'opzione 2. 2. Nella schermata Aggiungi demone alla lista, nel campo Nome demone, inserite un nome per il demone, quindi premete Invio.	Amministratore di sistema mainframe
Configurazione del publisher.	Per configurare l'editore, procedi come segue: 1. Nella schermata del menu principale di SQData z/OS, scegliete l'opzione 3. Verrà visualizzata la schermata	Amministratore di sistema mainframe

Operazione	Description	Competenze richieste
	di riepilogo di Capture/P ublisher. 2. Scegliete l'opzione per aggiungere un file CAB. Verrà visualizzata la schermata Aggiungi file CAB all'elenco. 3. Nel campo Nome, inserisci un nome per il file CAB. Per Db2, inserite Type as. `D` 4. Premere Invio. Verrà visualizzata la schermata Crea nuovo file CAB di Db2 Capture. 5. Nel campo ZfS Dir, specificate il punto di montaggio dello storage. 6. Premere Invio per salvare e continuare.	

Operazione	Description	Competenze richieste
Aggiorna il file di configurazione del demone.	Per aggiornare i dettagli dell'editore nel file di configurazione del demone del controller, procedi come segue: 1. Nella schermata del menu principale di SQData z/OS, scegliete l'opzione 2. 2. Entra `S` vicino al demone che hai creato per vedere i dettagli del demone. 3. Invio`1`, quindi premi Invio per modificare il file degli agenti. 4. Aggiungi i dettagli del file CAB. L'esempio seguente mostra i dettagli di un file CAB denominato. `DB2ZTOMSK` Usa il tuo ID utente del mainframe invece di. `<userid>` `ÝDB2ZTOMSK¨ type=capture cab=/u/<userid>/sqdata/DB2ZTOMSK.cab` 5. Premere F3. 6. Invio `2` per modificare il file ACL. Aggiungete il vostro `userid` al file di `acl` configurazione come mostrato nell'esempio seguente:	Amministratore di sistema mainframe

Operazione	Description	Competenze richieste
	`Ýacls¨ prod=admin,<userid>` 7. Premi F3 per salvare e uscire.	
Crea il job per avviare il daemon del controller.	Per creare il job, procedi come segue: 1. In Opzioni, immettereG. 2. Inserisci la JOB Card, le librerie job e proc e i dettagli `Db2 load` della biblioteca. 3. Immettete i dettagli del file ACL di rete e immettete l'opzione 2 per generare il file Job Control Language (JCL) nella libreria di job specificata.	Amministratore di sistema mainframe

Operazione	Description	Competenze richieste
Genera il file JCL di Capture Publisher.	Per generare il file JCL dell'editore di acquisizi one, effettuate le seguenti operazioni: 1. Nella schermata del menu principale di SQData z/OS, scegliete l'opzione 3. Verrà visualizzata la schermata di riepilogo di Capture/P ublisher. 2. Immettete `S` accanto al file CAB per selezionarlo. Verrà visualizzata la schermata Db2 Capture/Publisher Detail. 3. In Opzioni, immettete `G` le opzioni per generare il `capture/publisher` lavoro. 4. Immettete la scheda JOB, le librerie di job e procedure e i dettagli della libreria di caricamento Db2. 5. Per creare il lavoro, scegliete l'opzione 4. Il lavoro viene creato nella libreria dei lavori specificata nella libreria dei lavori.	Amministratore di sistema mainframe

Operazione	Description	Competenze richieste

Controlla e aggiorna CDC.	1. Controlla il DATACAPTURE flag della tabella Db2 eseguendo la seguente query, cambiando <table name> il nome della tabella Db2:
	`SELECT DATACAPTURE FROM SYSIBM.SYSTABLES WHERE NAME='<table name>';`
	Conferma che il risultato sia visualizzato comeDATACAPTURE . Y
	2. In caso DATACAPTURE contrarioY, esegui la seguente query per abilitare CDC sulla tabella Db2, cambiando <table name> il nome della tabella Db2:
	`ALTER TABLE <table name> DATA CAPTURE CHANGES;`
		Amministratore di sistema mainframe

Operazione	Description	Competenze richieste
Invia i file JCL.	Invia i seguenti file JCL che hai configurato nei passaggi precedenti: • File JCL per avviare il demone del controller • File JCL per avviare l'acquisizione e la pubblicazione Dopo aver inviato i file JCL, puoi avviare Apply Engine in Precisely sull' EC2 istanza.	Amministratore di sistema mainframe

Esegui e convalida CDC

Operazione	Description	Competenze richieste
Avvia Apply Engine e convalida il CDC.	Per avviare Apply Engine sull' EC2 istanza e convalidare il CDC, procedi come segue: 1. Per connetterti all' EC2 istanza, segui le istruzioni nella AWS documentazione. 2. Passa alla directory che contiene il `script.sqd` file: `cd /home/ec2-user/precisely/scripts`	Architetto del cloud, sviluppatore di app

Operazione	Description	Competenze richieste
	3. Per avviare Apply Engine, eseguite il seguente comando `sqdeng` start: `sqdeng -s script.sqd --identity=/home/ec2-user/.nacl/id_nacl` L'Apply Engine inizierà ad attendere gli aggiornamenti dal sorgente del mainframe. 4. Per testare CDC, inserite alcuni record o aggiorna la tabella Db2. 5. Verifica che il registro di Apply Engine mostri il numero di record acquisiti e scritti nella destinazione.	

Operazione	Description	Competenze richieste

Convalida i record sull'argomento Amazon MSK.	Per leggere il messaggio sull'argomento Kafka, procedi come segue:
	
	1. Per passare alla `bin` directory del percorso di installazione del client Kafka sull' EC2 istanza, esegui il comando seguente, sostituendolo `<Kafka version>` con la tua versione:
	
	`cd /home/ec2-user/kafka/kafka_2.13-<Kafka version>/bin`
	
	2. Per convalidare il CDC Db2 scritto come messaggi nell'argomento Kafka, esegui il comando seguente, sostituendo `<kafka broker>` e `<Topic Name>` con l'argomento creato in precedenza:
	
	`./kafka-console-consumer.sh --bootstrap-server <kafka broker>:9098 --topic <Topic Name> --from-beginning --consumer.config client.properties`
		Sviluppatore di app, architetto cloud

Operazione	Description	Competenze richieste
	3. Verifica che i messaggi corrispondano al numero di record aggiornati nella tabella Db2.	

Archivia i dati relativi alle modifiche del mainframe in un data warehouse Serverless Amazon Redshift

Operazione	Description	Competenze richieste
Configura Amazon Redshift Serverless.	Per creare un data warehouse Serverless Amazon Redshift, segui le istruzioni nella documentazione.AWS Nella dashboard di Amazon Redshift Serverless, verifica che lo spazio dei nomi e il gruppo di lavoro siano stati creati e siano disponibili. Per questo modello di esempio, il processo potrebbe richiedere 25 minuti.	Ingegnere dei dati
Configura il ruolo IAM e la policy di fiducia necessari per l'acquisizione dello streaming.	Per configurare l'acquisizione di streaming senza server di Amazon Redshift da Amazon MSK, procedi come segue: 1. Crea una policy IAM per Amazon Redshift per accedere ad Amazon MSK. Sostituendo `[region]` con Amazon MSK, `[account-id]` con il tuo Account AWS ID e `[msk-clus`	Ingegnere dei dati

Operazione	Description	Competenze richieste
	ter-name] con il nome del cluster Amazon MSK, esegui il codice seguente: Regione AWS `{"Version": "2012-10-17", "Statement": [{"Sid": "MSKIAMpolicy","Effect": "Allow","Action": ["kafka-cluster:ReadData","kafka-cluster:DescribeTopic","kafka-cluster:Connect"],"Resource": ["arn:aws:kafka:[region]:[account-id]:cluster/[msk-cluster-name]/*","arn:aws:kafka:[region]:[account-id]:topic/[msk-cluster-name]/*"]},{"Effect": "Allow","Action": ["kafka-cluster:AlterGroup","kafka-cluster:DescribeGroup"],"Resource": ["arn:aws:kafka:[region]:[account-id]:group/[msk-cluster-name]/*"]}]}`	

Operazione	Description	Competenze richieste
	Puoi trovare il nome del cluster e l'Amazon Resource Name (ARN) sulla console Amazon MSK. Sulla console, scegli Riepilogo del cluster, quindi scegli ARN. 2. Per creare un ruolo IAM e allegare la policy, segui le istruzioni nella AWS documentazione. 3. Per collegare il ruolo IAM allo spazio dei nomi Serverless di Amazon Redshift, procedi come segue: a. Accedi alla console e apri la console Amazon Redshift all'indirizzo. https://console.aws.amazon.com/redshiftv2/ b. Scegli serverless dashboard (Pannello di controllo serverless). c. Scegli Namespace. d. Scegli la scheda Sicurezza e crittografia. e. Scegli Autorizzazione e collega il ruolo IAM che hai creato.	

Operazione	Description	Competenze richieste
	4. Nel tuo gruppo di sicurezza Amazon Redshift Serverless, crea una regola in entrata con i seguenti dettagli: • Per Type (Tipo), scegliere Custom TCP (TCP personalizzato). • Per Protocol (Protocollo), selezionare TCP. • Per Port range, scegli 9098, 9198. • Per Source, scegli il gruppo di sicurezza Amazon MSK. 5. Nel tuo gruppo di sicurezza Amazon MSK, crea una regola in entrata con i seguenti dettagli: • Per Type (Tipo), scegliere Custom TCP (TCP personalizzato). • Per Protocol (Protocollo), selezionare TCP. • Per Port range, scegli 9098, 9198. • Per Source, scegli il gruppo di sicurezza Amazon Redshift. Questo modello utilizza la porta per l'autenticazione IAM per la configurazione di Amazon Redshift e	

Operazione	Description	Competenze richieste
	Amazon MSK. Per ulteriori informazioni, consulta la AWS documentazione (fase 2). 6. Attiva il routing VPC avanzato per il gruppo di lavoro Serverless Amazon Redshift. Per ulteriori informazioni, consulta la documentazione relativa ad AWS.	
Connetti Amazon Redshift Serverless ad Amazon MSK.	Per connetterti all'argomento Amazon MSK, crea uno schema esterno in Amazon Redshift Serverless. Nell'editor di query di Amazon Redshift v2, esegui il seguente comando SQL, sostituendolo `'iam_role_arn'` con il ruolo creato in precedenza e sostituendo `'MSK_cluster_arn'` «con l'ARN per il cluster. `CREATE EXTERNAL SCHEMA member_schema FROM MSK IAM_ROLE 'iam_role_arn' AUTHENTICATION iam URI 'MSK_cluster_arn';`	Ingegnere della migrazione

Operazione	Description	Competenze richieste
Crea una vista materializzata.	Per utilizzare i dati dell'argomento Amazon MSK in Amazon Redshift Serverless, crea una vista materializzata. Nell'editor di query di Amazon Redshift v2, esegui i seguenti comandi SQL, sostituendoli `<MSK_Topic_name>` con il nome del tuo argomento Amazon MSK. `CREATE MATERIALIZED VIEW member_view AUTO REFRESH YES AS SELECT kafka_partition, kafka_offset, refresh_time, json_parse(kafka_value) AS Data FROM member_schema.<MSK_Topic_name> WHERE CAN_JSON_PARSE(kafka_value);`	Ingegnere della migrazione

Operazione	Description	Competenze richieste

Crea tabelle di destinazione in Amazon Redshift.	Le tabelle Amazon Redshift forniscono l'input per Quick Sight. Questo modello utilizza le tabelle `member_dtls` e`member_plans` , che corrispondono alle tabelle Db2 di origine sul mainframe.
	
	Per creare le due tabelle in Amazon Redshift, esegui i seguenti comandi SQL in Amazon Redshift query editor v2:
	
	`-- Table 1: members_dtls`
	`CREATE TABLE members_dtls (`
	`memberid INT ENCODE AZ64,`
	`member_name VARCHAR(100) ENCODE ZSTD,`
	`member_type VARCHAR(50) ENCODE ZSTD,`
	`age INT ENCODE AZ64,`
	`gender CHAR(1) ENCODE BYTEDICT,`
	`email VARCHAR(100) ENCODE ZSTD,`
	`region VARCHAR(50) ENCODE ZSTD`
	`) DISTSTYLE AUTO;`
	
	`-- Table 2: member_plans`
	`CREATE TABLE member_plans (`
	`memberid INT ENCODE AZ64,`
		Ingegnere della migrazione

Operazione	Description	Competenze richieste

	<pre>medical_plan CHAR(1)
	ENCODE BYTEDICT,
	dental_plan CHAR(1)
	ENCODE BYTEDICT,
	vision_plan CHAR(1)
	ENCODE BYTEDICT,
	preventive_immuniz
	ation VARCHAR(50)
	ENCODE ZSTD
	) DISTSTYLE AUTO;</pre>	

Crea una procedura memorizzata in Amazon Redshift.	Questo modello utilizza una procedura memorizzata per sincronizzare i dati di modifica (`INSERT`,`UPDATE`,`DELETE`) dal mainframe di origine alla tabella del data warehouse Amazon Redshift di destinazi one per l'analisi in Quick Sight. Per creare la stored procedure in Amazon Redshift, utilizza l'editor di query v2 per eseguire il codice di stored procedure presente nel repository. GitHub	Ingegnere della migrazione

Operazione	Description	Competenze richieste
Leggi dalla vista materializzata in streaming e carica nelle tabelle di destinazione.	La stored procedure legge le modifiche ai dati dalla vista materializzata in streaming e carica le modifiche ai dati nelle tabelle di destinazione. Per eseguire la stored procedure , utilizzare il seguente comando: `call SP_Members_Load();` Puoi utilizzare Amazon EventBridge per pianifica re i lavori nel tuo data warehouse Amazon Redshift per richiamare questa procedura memorizzata in base ai tuoi requisiti di latenza dei dati. EventBridge esegue i lavori a intervalli fissi. Per verificare se la chiamata precedente alla procedura è stata completata, potrebbe essere necessario utilizzar e un meccanismo come una macchina a AWS Step Functionsstati. Per maggiori informazioni, consulta le seguenti risorse: • Creazione di una EventBrid ge regola Amazon che viene eseguita secondo una pianificazione	Ingegnere della migrazione

Operazione	Description	Competenze richieste
	• Accelera l'orchestrazione di un processo ELT utilizzando Amazon AWS Step Functions Redshift Data API. Un'altra opzione è usare Amazon Redshift Query Editor v2 per pianificare l'aggiornamento. Per ulteriori informazioni, consulta Pianificazione di una query con l'editor di query v2.	

Connect Quick Sight ai dati in Amazon Redshift

Operazione	Description	Competenze richieste
Configura Quick Sight.	Per configurare Quick Sight, segui le istruzioni nella AWS documentazione.	Ingegnere della migrazione
Configura una connessione sicura tra Quick Sight e Amazon Redshift.	Per configurare una connessione sicura tra Quick Sight e Amazon Redshift, procedi come segue 1. Per autorizzare le connessioni da Quick Sight ad Amazon Redshift, apri la console Amazon Redshift e aggiungi una regola in entrata nel gruppo di sicurezza Amazon Redshift. La regola	Ingegnere della migrazione

Operazione	Description	Competenze richieste
	dovrebbe consentire il traffico verso la porta 5439 (la porta Redshift predefinita) dall'intervallo CIDR in cui è impostato Quick Sight. Per un elenco dei relativi indirizzi IP, consulta Regioni AWS Supported Regioni AWS for Quick Sight. Sulla console Amazon Redshift, scegli Gruppo di lavoro, Accesso ai dati, Rete e sicurezza e abilita Accessibile pubblicamente.	

Operazione	Description	Competenze richieste
Crea un set di dati per Quick Sight.	Per creare un set di dati per Quick Sight da Amazon Redshift, procedi come segue: 1. Nella console Quick Sight, nel pannello di navigazione, scegli Datasets. 2. Nella pagina Set di dati scegli Nuovo set di dati. 3. Scegli Redshift Manual connect. 4. Nella finestra Nuova origine dati Redshift, inserisci le informazioni di connessio ne: • Per Nome origine dati, inserisci un nome per l'origine dati Amazon Redshift. • Per Database server, inserisci l'endpoint del cluster Amazon Redshift. Puoi ottenere il valore dell'endpoint dal campo Endpoint nella sezione Informazioni generali per il gruppo di lavoro del cluster sulla dashboard di Amazon Redshift Serverless. L'indirizzo del server è la prima parte dell'endpoint prima dei due punti, come mostrato nell'esempio seguente:	Tecnico di migrazione

Operazione	Description	Competenze richieste
	`mfdata-insights.NNNNNNNN.us-east-1.redshift-serverless.amazonaws.com:5439/dev` • Per la porta, inserisci `5439` (la porta predefinita per Amazon Redshift). • Inserisci il nome del database (dopo la barra nell'endpoint). In questo caso, il nome del database è. `dev` • Per Nome utente e password, inserisci il nome utente e la password per il database Amazon Redshift. 5. Scegli Convalida connessione. In caso di successo, dovresti vedere un segno di spunta verde, che indica la convalida. Se la convalida fallisce, consulta la sezione Risoluzione dei problemi. 6. Seleziona Create data source (Crea origine dati).	

Operazione	Description	Competenze richieste
Unisciti al set di dati.	Per creare analisi in Quick Sight, unisci le due tabelle seguendo le istruzioni nella AWS documentazione. Nel riquadro Join Configura tion, scegli Left for Join type. In Clausole Join, usa. `memberid from member_plans = memberid from members_details`	Ingegnere della migrazione

Ottieni informazioni aziendali dai dati del mainframe utilizzando Amazon Q in Quick Sight

Operazione	Description	Competenze richieste
Configura Amazon Q in Quick Sight.	Per configurare la funzional ità di BI generativa di Amazon Q in Quick Sight, segui le istruzioni nella AWS documentazione.	Ingegnere della migrazione
Analizza i dati del mainframe e crea una dashboard visiva.	Per analizzare e visualizzare i dati in Quick Sight, procedi come segue: 1. Per creare l'analisi dei dati del mainframe, segui le istruzioni nella AWS documentazione. Per il set di dati, scegli il set di dati che hai creato. 2. Nella pagina di analisi scegli Build visual.	Ingegnere della migrazione

Operazione	Description	Competenze richieste
	3. Nella finestra Crea argomento per l'analisi, scegli Aggiorna argomento esistente. 4. Nell'elenco a discesa Seleziona un argomento, scegli l'argomento che hai creato in precedenza. 5. Scegli Collegamento all'argomento. 6. Dopo aver collegato l'argomento, scegli Build visual per aprire la finestra Amazon Q Build a Visual. 7. Nella barra delle istruzioni, scrivi le tue domande di analisi. Le domande di esempio utilizzate per questo modello sono le seguenti: • Mostra la distribuzione dei membri per regione • Mostra la distribuzione dei membri per età • Mostra la distribuzione dei membri per genere • Mostra la distribuzione dei membri per tipo di piano • Mostra membri che non hanno completato l'immunizzazione preventiva	

Operazione	Description	Competenze richieste
	Dopo aver inserito le domande, scegli Costruisci. Amazon Q in Quick Sight crea le immagini. 8. Per aggiungere gli elementi visivi alla tua dashboard visiva, scegli AGGIUNGI ALL'ANALISI. Al termine, puoi pubblicare la dashboard per condividerla con altri membri dell'organizzazione. Ad esempio, consulta la dashboard visiva di Mainframe nella sezione Informazioni aggiuntive.	

Crea una storia di dati con Amazon Q in Quick Sight a partire dai dati mainframe

Operazione	Description	Competenze richieste
Crea una storia basata sui dati.	Crea una storia basata sui dati per spiegare gli approfondimenti tratti dall'analisi precedente e genera una raccomandazione per aumentare l'immunizzazione preventiva per i membri: 1. Per creare la storia dei dati, segui le istruzioni nella AWS documentazione.	Ingegnere della migrazione

Operazione	Description	Competenze richieste
	2. Per il prompt della storia dei dati, usa quanto segue: `Build a data story about Region with most numbers of members. Also show the member distribution by medical plan, vision plan, dental plan. Recommend how to motivate members to complete immunizat ion. Include 4 points of supportin g data for this pattern.` Puoi anche creare un prompt personalizzato per generare storie basate sui dati per altri approfond imenti aziendali. 3. Scegli Aggiungi elementi visivi e aggiungi gli elementi visivi pertinenti alla storia dei dati. Per questo modello, usa le immagini che hai creato in precedenz a. 4. Seleziona Crea. 5. Ad esempio, l'output di una storia di dati, vedi Data	

Operazione	Description	Competenze richieste
	story output nella sezione Informazioni aggiuntive.	
Visualizza la storia dei dati generati.	Per visualizzare la storia generata dai dati, scegli quella storia nella pagina Storie di dati.	Ingegnere della migrazione
Modifica una storia di dati generata.	Per modificare la formattazione, il layout o gli elementi visivi di una storia di dati, segui le istruzioni contenute nella AWS documentazione.	Ingegnere della migrazione
Condividi una storia basata sui dati.	Per condividere una storia basata sui dati, segui le istruzioni nella AWS documentazione.	Ingegnere della migrazione

risoluzione dei problemi

Problema	Soluzione
La creazione del set di dati da Quick Sight ad Amazon Redshift non è riuscita. `Validate Connection`	1. Verifica che il gruppo di sicurezza collegato all'istanza Serverless di Amazon Redshift consenta il traffico in entrata dall'intervallo di indirizzi IP associato alla regione in cui hai configurato Quick Sight. 2. Verifica che il VPC su cui è distribuito Amazon Redshift Serverless sia disponibile pubblicamente. 3. Conferma di utilizzare il nome utente e la password corretti per Amazon Redshift. Puoi

Problema	Soluzione
	reimpostare il nome utente e la password sulla console Amazon Redshift.
Il tentativo di avviare il motore Apply sull' EC2 istanza restituisce il seguente errore: `-bash: sqdeng: command not found`	Esporta il percorso di `sqdata` installazione eseguendo il seguente comando: `export PATH=$PATH:/usr/sbin:/opt/precisely/di/sqdata/bin`
Il tentativo di avviare Apply Engine restituisce uno dei seguenti errori di connessione: • `SQDD018E Cannot connect to transfer socket(rc==0x18468). Agent:<Agent Name > Socket:/u/./sqdata/.DB2ZTOMSK.cab.data` • `SQDUR06E Error opening url cdc://<VPC end point name>:2626/DB2ZTOMSK/DB2ZTOMSK : errno:1128 (Unknown error 1128)`	Controllate lo spool del mainframe per assicurarvi che i daemon job del controller siano in esecuzione.

Risorse correlate

- Genera approfondimenti utilizzando Modernizzazione del mainframe AWS Amazon Q in Quick Sight (modello)
- Genera approfondimenti sui dati utilizzando Modernizzazione del mainframe AWS Amazon Q in Quick Sight (demo)
- Modernizzazione del mainframe AWS - Replica dei dati per IBM z/OS
- Inserimento dello streaming da Amazon Redshift in una vista materializzata

Informazioni aggiuntive

Esempio di file.ddl

members_details.ddl

```
CREATE TABLE MEMBER_DTLS (
memberid INTEGER NOT NULL,
member_name VARCHAR(50),
member_type VARCHAR(20),
age INTEGER,
gender CHAR(1),
email VARCHAR(100),
region VARCHAR(20)
);
```

piani_membri.ddl

```
CREATE TABLE MEMBER_PLANS (
memberid INTEGER NOT NULL,
medical_plan CHAR(1),
dental_plan CHAR(1),
vision_plan CHAR(1),
preventive_immunization VARCHAR(20)
);
```

Esempio di file.sqd

Sostituisci **<kafka topic name>**con il nome dell'argomento Amazon MSK.

script.sqd

```
-- Name: DB2ZTOMSK: DB2z To MSK JOBNAME DB2ZTOMSK;REPORT EVERY 1;OPTIONS
 CDCOP('I','U','D');-- Source Descriptions
JOBNAME DB2ZTOMSK;
REPORT EVERY 1;
OPTIONS CDCOP('I','U','D');

-- Source Descriptions
BEGIN GROUP DB2_SOURCE;
DESCRIPTION DB2SQL /var/precisely/di/sqdata/apply/DB2ZTOMSK/ddl/mem_details.ddl AS
 MEMBER_DTLS;
DESCRIPTION DB2SQL /var/precisely/di/sqdata/apply/DB2ZTOMSK/ddl/mem_plans.ddl AS
 MEMBER_PLANS;
END GROUP;
-- Source Datastore
DATASTORE cdc://<zos_host_name>/DB2ZTOMSK/DB2ZTOMSK
OF UTSCDC
```

```
AS CDCIN
DESCRIBED BY GROUP DB2_SOURCE ;
-- Target Datastore(s)
DATASTORE 'kafka:///<kafka topic name>/key'
OF JSON
AS TARGET
DESCRIBED BY GROUP DB2_SOURCE;
PROCESS INTO TARGET
SELECT
{
REPLICATE(TARGET)
}
FROM CDCIN;
```

Dashboard visivo del mainframe

La seguente immagine dei dati è stata creata da Amazon Q in Quick Sight per la domanda di analisi `show member distribution by region`.

La seguente immagine dei dati è stata creata da Amazon Q in Quick Sight per la domanda`show member distribution by Region who have not completed preventive immunization, in pie chart`.

Produzione della storia dei dati

Le schermate seguenti mostrano sezioni della storia dei dati creata da Amazon Q in Quick Sight per il prompt`Build a data story about Region with most numbers of members. Also show the member distribution by age, member distribution by gender. Recommend how to motivate members to complete immunization. Include 4 points of supporting data for this pattern`.

Nell'introduzione, la storia dei dati consiglia di scegliere la regione con il maggior numero di membri per ottenere il massimo impatto dagli sforzi di immunizzazione.

La storia dei dati fornisce un'analisi del numero di membri per le quattro regioni. Le regioni del Nordest, del Sud-Ovest e del Sud-Est hanno il maggior numero di membri.

La storia dei dati presenta un'analisi dei membri per età.

La storia dei dati si concentra sugli sforzi di immunizzazione nel Midwest.

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Genera approfondimenti sui dati utilizzando Modernizzazione del mainframe AWS Amazon Q in Quick Sight

Shubham Roy, Roshna Razack e Santosh Kumar Singh, Amazon Web Services

Riepilogo

Nota: il Modernizzazione del mainframe AWS servizio (esperienza Managed Runtime Environment) non è più aperto a nuovi clienti. Per funzionalità simili a Modernizzazione del mainframe AWS Service (Managed Runtime Environment experience), esplora Modernizzazione del mainframe AWS Service (Self-Managed Experience). I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta Modifica della Modernizzazione del mainframe AWS disponibilità.

Se la tua organizzazione ospita dati aziendali critici in un ambiente mainframe, ottenere informazioni approfondite da tali dati è fondamentale per promuovere la crescita e l'innovazione. Sbloccando i dati del mainframe, puoi creare business intelligence più veloce, sicura e scalabile per accelerare il processo decisionale, la crescita e l'innovazione basati sui dati nel cloud Amazon Web Services ().AWS

Questo modello presenta una soluzione per generare informazioni aziendali e creare narrazioni condivisibili dai dati mainframe utilizzando il trasferimento di Modernizzazione del mainframe AWS file con BMC e Amazon Q in Quick Sight. I set di dati mainframe vengono trasferiti su Amazon Simple Storage Service (Amazon S3) utilizzando Modernizzazione del mainframe AWS il trasferimento di file con BMC. Una AWS Lambda funzione formatta e prepara il file di dati del mainframe per il caricamento in Quick Sight.

Dopo che i dati sono disponibili in Quick Sight, puoi utilizzare istruzioni in linguaggio naturale con Amazon Q in Quick Sight per creare riepiloghi dei dati, porre domande e generare storie di dati. Non è necessario scrivere query SQL o imparare a usare uno strumento di business intelligence (BI).

Contesto aziendale

Questo modello presenta una soluzione per i casi d'uso di analisi dei dati mainframe e analisi dei dati. Utilizzando il modello, crei una dashboard visiva per i dati della tua azienda. Per dimostrare la soluzione, questo modello utilizza un'azienda sanitaria che fornisce piani medici, dentistici e oculistici ai suoi membri negli Stati Uniti. In questo esempio, i dati demografici dei membri e le informazioni sul piano vengono archiviati nei set di dati del mainframe. La dashboard visiva mostra quanto segue:

- Distribuzione dei membri per regione

- Distribuzione dei membri per genere
- Distribuzione dei membri per età
- Distribuzione dei membri per tipo di piano
- Membri che non hanno completato l'immunizzazione preventiva

Dopo aver creato la dashboard, si genera una storia basata sui dati che spiega le informazioni ricavate dall'analisi precedente. La storia dei dati fornisce raccomandazioni per aumentare il numero di membri che hanno completato le vaccinazioni preventive.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Set di dati mainframe con dati aziendali
- Accesso per installare un agente di trasferimento file sul mainframe

Limitazioni

- Il file di dati del mainframe deve essere in uno dei formati di file supportati da Quick Sight. Per un elenco dei formati di file supportati, consultate Fonti di dati supportate.
- Questo modello utilizza una funzione Lambda per convertire il file mainframe in un formato supportato da Quick Sight.

Architecture

Il diagramma seguente mostra un'architettura per generare informazioni aziendali dai dati mainframe utilizzando il trasferimento di Modernizzazione del mainframe AWS file con BMC e Amazon Q in Quick Sight.

Il diagramma mostra il flusso di lavoro seguente:

1. Un set di dati mainframe contenente dati aziendali viene trasferito su Amazon S3 Modernizzazione del mainframe AWS utilizzando il trasferimento di file con BMC.
2. La funzione Lambda converte il file contenuto nel bucket S3 di destinazione del trasferimento file in formato CSV (valori separati da virgole).

3. La funzione Lambda invia il file convertito al bucket S3 del set di dati di origine.
4. I dati contenuti nel file vengono acquisiti da Quick Sight.
5. Gli utenti accedono ai dati in Quick Sight. Puoi utilizzare Amazon Q in Quick Sight per interagire con i dati utilizzando istruzioni in linguaggio naturale.

Tools (Strumenti)

Servizi AWS

- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Modernizzazione del mainframe AWS il trasferimento di file con BMC converte e trasferisce i set di dati mainframe su Amazon S3 per casi d'uso di modernizzazione, migrazione e potenziamento del mainframe.
- Amazon Quick Sight è un servizio di BI su scala cloud che ti aiuta a visualizzare, analizzare e generare report sui dati in un'unica dashboard. Questo modello utilizza le funzionalità di BI generativa di Amazon Q in Quick Sight.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Best practice

- Quando crei i ruoli AWS Identity and Access Management (IAM) per Modernizzazione del mainframe AWS il trasferimento di file con BMC e la funzione Lambda, segui il principio del privilegio minimo.
- Assicurati che il set di dati di origine supporti i tipi di dati per Quick Sight. Se il set di dati di origine contiene tipi di dati non supportati, convertili in tipi di dati supportati. Per informazioni sui tipi di dati mainframe non supportati e su come convertirli in tipi di dati supportati da Amazon Q in Quick Sight, consulta la sezione Risorse correlate.

Epiche

Configura il trasferimento di Modernizzazione del mainframe AWS file con BMC

Operazione	Description	Competenze richieste
Installa l'agente di trasferim ento file.	Per installare l'agente di trasferimento Modernizzazione del mainframe AWS file, segui le istruzioni nella AWS documentazione.	Amministratore di sistema mainframe
Crea un bucket S3 per il trasferimento di file su mainframe.	Crea un bucket S3 per archiviare il file di output del trasferimento di file con BMC. Modernizzazione del mainframe AWS Nel diagramma di architettura, questo è il bucket di destinazi one per il trasferimento dei file.	Ingegnere della migrazione
Crea l'endpoint per il trasferim ento dei dati.	1. Crea un bucket S3 per lo staging del file mainframe di input per Modernizz azione del mainframe AWS il trasferimento di file con BMC. 2. Per creare l'endpoin t di trasferimento dati mainframe, segui le istruzioni contenute nella documentazione.AWS	Specialista in modernizz azione dei mainframe di AWS

Converti l'estensione del nome del file mainframe per l'integrazione con Quick Sight

Operazione	Description	Competenze richieste
Crea un bucket S3.	Crea un bucket S3 per la funzione Lambda per copiare il file mainframe convertito dal bucket di origine a quello di destinazione finale.	Ingegnere della migrazione
Creazione di una funzione Lambda.	Per creare una funzione Lambda che modifichi l'estensione del file e copi il file mainframe nel bucket di destinazione, procedi come segue: 1. Accedi a e Console di gestione AWS accedi alla console. AWS Lambda 2. Scegli Crea funzione, quindi scegli Autore da zero. 3. Per Nome funzione, inserisci un nome per la tua funzione. 4. Nell'elenco a discesa Runtime, scegli Python.3.X. 5. Espandi Modifica ruolo di esecuzione predefinito, quindi scegli Crea un nuovo ruolo con autorizzazioni Lambda di base. 6. Scegli Crea funzione. 7. Scegli la scheda Codice, quindi incolla il codice `S3CopyLambda.py`	Ingegnere della migrazione

Operazione	Description	Competenze richieste
	Python fornito nella sezione Informazioni aggiuntive. Il codice Python è stato generato utilizzando Amazon Q nell'ambiente di sviluppo integrato (IDE) di Microsoft Visual Studio. 8. Modifica `destination_bucket_name` il nome del bucket S3 creato in precedenza e `change destination_file_key` il nome del file mainframe. 9. Distribuire la funzione Lambda.	

Operazione	Description	Competenze richieste
Crea un trigger Amazon S3 per richiamare la funzione Lambda.	Per configurare un trigger che richiama la funzione Lambda, procedi come segue: 1. Sulla console Lambda, apri la pagina Funzioni. 2. Scegli la funzione Lambda. 3. Nella panoramica delle funzioni, scegli Aggiungi trigger. 4. Nell'elenco a discesa di configurazione Trigger, scegli S3. 5. Nel campo Bucket, inserisci il nome del tuo bucket di origine. 6. Nell'elenco a discesa Tipo di evento, scegli Tutti gli oggetti, crea eventi. 7. Seleziona la casella di controllo Riconosco che l'utilizzo dello stesso bucket S3 sia per l'input che per l'output non è consigliato, quindi scegli Aggiungi. Per maggiori informazioni consulta Tutorial: uso di un trigger Amazon S3 per richiamare una funzione Lambda.	Responsabile della migrazione

Operazione	Description	Competenze richieste
Fornisci le autorizzazioni IAM per la funzione Lambda.	Le autorizzazioni IAM sono necessarie per consentir e alla funzione Lambda di accedere ai bucket S3 del set di dati di origine e di destinazione del trasferim ento di file. Aggiorna la policy associata al ruolo di esecuzion e della funzione Lambda autorizzando `s3:GetObj ect` e `s3:DeleteObject` autorizzando il bucket S3 di destinazione per il trasferim ento di file e l'`s3:PutObj ect` accesso per il bucket S3 del set di dati di origine. Per ulteriori informazioni, consulta la sezione Creare una politica di autorizzazioni in Tutorial: Usare un trigger di Amazon S3 per richiamare una funzione Lambda.	Responsabile della migrazione

Definisci un'attività di trasferimento dati su mainframe

Operazione	Description	Competenze richieste
Crea un'attività di trasferim ento per copiare il file mainframe nel bucket S3.	Per creare un'operazione di trasferimento di file su mainframe, segui le istruzion i nella documentazione.Mod ernizzazione del mainframe AWS	Ingegnere della migrazione

Operazione	Description	Competenze richieste
	Note Specificate la codifica della tabella codici sorgente come IBM1047 e la codifica della tabella codici di destinazione come UTF-8.	
Verifica l'attività di trasferimento.	Per verificare che il trasferimento dei dati abbia esito positivo, segui le istruzioni contenute nella Modernizzazione del mainframe AWS documentazione. Verificate che il file mainframe si trovi nel bucket S3 di destinazione per il trasferimento dei file.	Responsabile della migrazione
Verifica la funzione di copia Lambda.	Verificate che la funzione Lambda sia avviata e che il file venga copiato con un'estensione.csv nel bucket S3 del set di dati di origine. Il file.csv creato dalla funzione Lambda è il file di dati di input per Quick Sight. Per i dati di esempio, consulta il `Sample-data-member-healthcare-APG` file nella sezione Allegati.	Responsabile della migrazione

Connect Quick Sight ai dati del mainframe

Operazione	Description	Competenze richieste
Configura Quick Sight.	Per configurare Quick Sight, segui le istruzioni nella AWS documentazione.	Responsabile della migrazione
Crea un set di dati per Quick Sight.	Per creare un set di dati per Quick Sight, segui le istruzioni nella AWS documentazione. Il file di dati di input è il file mainframe convertito che è stato creato quando è stata definita l'attività di trasferimento dei dati del mainframe.	Responsabile della migrazione

Ottieni informazioni aziendali dai dati del mainframe utilizzando Amazon Q in Quick Sight

Operazione	Description	Competenze richieste
Configura Amazon Q in Quick Sight.	Questa funzionalità richiede Enterprise Edition. Per configurare Amazon Q in Quick Sight, procedi come segue: 1. Per ottenere il componente aggiuntivo Amazon Q, segui le istruzioni Passaggio 1: Scarica il componente aggiuntivo Q nella AWS documentazione. 2. Per utilizzare le funzionalità di BI generativa di Amazon Q, aggiorna gli account dei tuoi utenti. Segui le istruzion	Responsabile della migrazione

Operazione	Description	Competenze richieste
	i contenute nella documentazione.AWS 3. Crea un argomento Amazon Q utilizzando il set di dati creato in precedenza. Segui le istruzioni contenute nella AWS documentazione. 4. Per configurare i metadati degli argomenti in modo che siano compatibili con il linguaggio naturale, segui le istruzioni nella documentazione.AWS	

Operazione	Description	Competenze richieste
Analizza i dati del mainframe e crea una dashboard visiva.	Per analizzare e visualizzare i dati in Quick Sight, procedi come segue: 1. Per creare l'analisi dei dati del mainframe, segui le istruzioni nella AWS documentazione. Per il set di dati, scegli il set di dati creato nel passaggio precedente. 2. Nella pagina di analisi scegli Build visual. 3. Nella finestra Crea argomento per l'analisi, scegli Aggiorna argomento esistente. 4. Nell'elenco a discesa Seleziona un argomento, scegli l'argomento che hai creato in precedenza. 5. Scegli Collegamento all'argomento. 6. Dopo aver collegato l'argomento, scegli Build visual per aprire la finestra Amazon Q Build a Visual. 7. Nella barra delle istruzioni, scrivi le tue domande di analisi. Le domande di esempio utilizzate per questo modello sono le seguenti:	Ingegnere della migrazione

Operazione	Description	Competenze richieste
	• Mostra la distribuzione dei membri per regione • Mostra la distribuzione dei membri per età • Mostra la distribuzione dei membri per genere • Mostra la distribuzione dei membri per tipo di piano • Mostra l'immunizzazione preventiva del membro che non ha completato Dopo aver inserito le domande, scegli Costruisci. Amazon Q in Quick Sight crea le immagini. 8. Per aggiungere gli elementi visivi alla tua dashboard visiva, scegli AGGIUNGI ALL'ANALISI. Al termine, puoi pubblicare la dashboard per condividerla con altri membri dell'organizzazione. Ad esempio, consulta la dashboard visiva di Mainframe nella sezione Informazioni aggiuntive.	

Crea una storia di dati con Amazon Q in Quick Sight a partire dai dati del mainframe

Operazione	Description	Competenze richieste
Crea una storia basata sui dati.	Crea una storia basata sui dati per spiegare gli approfondimenti tratti dall'analisi precedente e genera una raccomandazione per aumentare l'immunizzazione preventiva per i membri: 1. Per creare la storia dei dati, segui le istruzioni nella AWS documentazione. 2. Per il prompt della storia dei dati, usa quanto segue: `Build a data story about Region with most numbers of members. Also show the member distribution by medical plan, vision plan, dental plan. Recommend how to motivate members to complete immunization. Include 4 points of supporting data for this pattern.` Puoi anche creare un prompt personalizzato per generare storie basate	Ingegnere della migrazione

Operazione	Description	Competenze richieste
	sui dati per altri approfond imenti aziendali. 3. Scegli Aggiungi elementi visivi e aggiungi gli elementi visivi pertinenti alla storia dei dati. Per questo modello, usa le immagini che hai creato in precedenz a. 4. Seleziona Crea. 5. Ad esempio, l'output di una storia di dati, vedi Data story output nella sezione Informazioni aggiuntive.	
Visualizza la storia dei dati generati.	Per visualizzare la storia dei dati generata, segui le istruzioni nella AWS documentazione.	Responsabile della migrazione
Modifica una storia di dati generata.	Per modificare la formattaz ione, il layout o gli elementi visivi di una storia di dati, segui le istruzioni contenute nella AWS documentazione.	Responsabile della migrazione
Condividi una storia basata sui dati.	Per condividere una storia basata sui dati, segui le istruzioni nella AWS documentazione.	Ingegnere della migrazione

risoluzione dei problemi

Problema	Soluzione
Impossibile individuare i file o i set di dati del mainframe inseriti nei criteri di ricerca dei set di dati per Crea attività di trasferimento in Modernizzazione del mainframe AWS file transfer with BMC.	1. Innanzitutto, verifica la connessione selezionando Data transfer endpoint sulla console Modernizzazione del mainframe AWS Transfer with BMC. Se la durata dell'ultimo battito cardiaco è superiore a due minuti, la connessione per il trasferim ento dei file non è stata stabilita. Se l'ultimo tempo di heartbeat è inferiore a 2 minuti per l'agente in esecuzione sul mainframe, la connessione all'agente ha esito positivo. Procedere al passaggio 2. 2. Controlla la Gestione dei segreti AWS configurazione. Una chiave segreta deve essere configurata in Secrets Manager con una chiave `userId` (I maiuscola) con un valore dell'ID utente del mainframe e una chiave `password` con il valore della password del mainframe. Le chiavi `password` segrete fanno distinzione tra maiuscole `userId` e minuscole e devono essere immesse così come sono.

Risorse correlate

Per convertire tipi di dati mainframe come PACKED-DECIMAL (COMP-3) o BINARY (COMP o COMP-4) in un tipo di dati supportato da Quick Sight, osservate i seguenti schemi:

- Converti e decomprimi i dati EBCDIC in ASCII usando Python AWS
- Converti i file mainframe dal formato EBCDIC al formato ASCII delimitato da caratteri in Amazon S3 utilizzando AWS Lambda

Informazioni aggiuntive

CopyLambdaS3 .py

Il seguente codice Python è stato generato utilizzando un prompt con Amazon Q in un IDE:

```
#Create a lambda function triggered by S3. display the S3 bucket name and key
import boto3
s3 = boto3.client('s3')
def lambda_handler(event, context):
print(event)
bucket = event['Records'][0]['s3']['bucket']['name']
key = event['Records'][0]['s3']['object']['key']
print(bucket, key)
#If key starts with object_created, skip copy, print "copy skipped". Return lambda with
 key value.
if key.startswith('object_created'):
print("copy skipped")
return {
'statusCode': 200,
'body': key
}
# Copy the file from the source bucket to the destination bucket.
 Destination_bucket_name = 'm2-filetransfer-final-opt-bkt'. Destination_file_key =
 'healthdata.csv'
copy_source = {'Bucket': bucket, 'Key': key}
s3.copy_object(Bucket='m2-filetransfer-final-opt-bkt', Key='healthdata.csv',
 CopySource=copy_source)
print("file copied")
#Delete the file from the source bucket.
s3.delete_object(Bucket=bucket, Key=key)
return {
'statusCode': 200,
'body': 'Copy Successful'
}
```

Pannello di controllo visivo del mainframe

La seguente immagine dei dati è stata creata da Amazon Q in Quick Sight per la domanda di analisi `show member distribution by region`.

La seguente immagine dei dati è stata creata da Amazon Q in Quick Sight per la domanda`show member distribution by Region who have not completed preventive immunization, in pie chart.`

Output della storia dei dati

Le schermate seguenti mostrano sezioni della storia dei dati creata da Amazon Q in Quick Sight per il prompt. `Build a data story about Region with most numbers of members. Also show the member distribution by medical plan, vision plan, dental plan. Recommend how to motivate members to complete immunization. Include 4 points of supporting data.`

Nell'introduzione, la storia dei dati consiglia di scegliere la regione con il maggior numero di membri per ottenere il massimo impatto dagli sforzi di immunizzazione.

La storia dei dati fornisce un'analisi del numero di membri delle tre principali regioni e nomina il sud-ovest come la regione principale per quanto riguarda gli sforzi di immunizzazione.

> **Note**
>
> Le regioni del sud-ovest e del nord-est hanno ciascuna otto membri. Tuttavia, il sud-ovest ha più membri che non sono completamente vaccinati, quindi ha maggiori possibilità di beneficiare delle iniziative per aumentare i tassi di immunizzazione.

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Implementa l'autenticazione basata su Microsoft Entra ID in un'applicazione mainframe modernizzata AWS Blu Age

Vishal Jaswani e Rimpy Tewani, Amazon Web Services

Riepilogo

> **Note**
>
> Modernizzazione del mainframe AWS Service (Managed Runtime Environment experience) non è più aperto a nuovi clienti. Per funzionalità simili a Modernizzazione del mainframe AWS Service (Managed Runtime Environment experience), esplora Modernizzazione del mainframe AWS Service (Self-Managed Experience). I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta Modifica della Modernizzazione del mainframe AWS disponibilità.

Le applicazioni mainframe modernizzate utilizzando modelli di refactoring, come quelle di Modernizzazione del mainframe AWS Refactor con AWS Blu Age, richiedono un'attenta integrazione dei meccanismi di autenticazione nella nuova architettura applicativa. Questa integrazione viene in genere considerata un'attività di post-modernizzazione. L'attività può essere complessa e spesso comporta la migrazione o l'esternalizzazione dei sistemi di autenticazione esistenti per allinearli ai moderni standard di sicurezza e alle pratiche native del cloud. Gli sviluppatori devono considerare come implementare l'autenticazione in modo efficace mentre lavorano entro i limiti dell'ambiente di runtime e delle librerie dell'applicazione modernizzata. Dopo la modernizzazione, AWS offre modi per semplificare l'integrazione del codice moderno AWS Blu Age con sistemi di gestione delle identità e degli accessi come Amazon Cognito e Microsoft Entra ID (precedentemente noto come Azure AD).

Questo modello spiega come implementare un meccanismo di autenticazione nell'applicazione modernizzata quando il provider di autenticazione è Microsoft Entra ID, senza perdere tempo in ricerche e prove. Il modello prevede:

- Librerie Angular testate sul campo e pertinenti della Microsoft Authentication Library (MSAL) e altra documentazione Microsoft Entra ID che sono essenziali per l'implementazione dell'autenticazione.
- Configurazioni richieste su AWS Blu Age Runtime per abilitare Spring Security utilizzando 2.0. OAuth
- Una libreria che acquisisce le identità degli utenti autenticati e le trasmette a Blu Age Runtime. AWS

- Misure di sicurezza che consigliamo di implementare.
- Suggerimenti per la risoluzione dei problemi più comuni relativi alla configurazione dell'ID Microsoft Entra.

> **Note**
>
> Questo modello utilizza la libreria di OAuth estensioni AWS Blu Age, fornita ai clienti nell'ambito dei servizi AWS professionali. Questa libreria non fa parte di AWS Blu Age Runtime.

Prerequisiti e limitazioni

Prerequisiti

- Un'applicazione modernizzata prodotta dagli strumenti di refactoring per la modernizzazione del mainframe AWS Blu Age. Questo modello utilizza CardDemocome esempio un'applicazione mainframe open source.
- La libreria di OAuth estensioni AWS Blu Age, fornita dal team AWS Blu Age durante la collaborazione con AWS Professional Services.
- Un attivo Account AWS per implementare e testare l'applicazione modernizzata.
- Familiarità con i file di configurazione AWS Blu Age e i fondamenti dell'ID Microsoft Entra.

Limitazioni

- Questo modello copre l'autenticazione OAuth 2.0 e i flussi di autorizzazione di base basati su token. Gli scenari di autorizzazione avanzati e i meccanismi di controllo degli accessi dettagliati non rientrano nell'ambito di applicazione.
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Versioni del prodotto

Questo modello è stato sviluppato utilizzando:

- AWS Blu Age Runtime versione 4.1.0 (il pattern funziona anche con le versioni successive che sono retrocompatibili)
- Libreria MSAL versione 3.0.23
- Java Development Kit (JDK) versione 17
- Versione Angular 16.1

Architecture

Stack tecnologico di origine

Nei tipici ambienti mainframe, l'autenticazione viene implementata tramite profili utente. Questi profili identificano gli utenti del sistema, definiscono chi può accedere e specificano quali funzioni gli utenti possono eseguire sulle risorse di sistema. I profili utente sono gestiti dai responsabili della sicurezza o dagli amministratori della sicurezza.

Stack tecnologico Target

- ID Microsoft Entra
- Backend modernizzato basato su Java Spring Boot
- AWS Blu Age Runtime
- Spring Security con OAuth 2.0
- Applicazione angolare a pagina singola (SPA)

Architettura Target

AWS Il runtime Blu Age supporta l'autenticazione OAuth basata su 2.0 per impostazione predefinita, quindi il pattern utilizza quello standard per proteggere il backend APIs.

Il diagramma seguente illustra il flusso del processo.

> **Note**
>
> Il diagramma include Amazon Aurora come esempio di modernizzazione del database, sebbene Aurora non sia inclusa nei passaggi di questo modello.

dove:

1. Un utente tenta di autenticarsi con l'ID Microsoft Entra.
2. Microsoft Entra ID restituisce i token di aggiornamento, accesso e ID utilizzati dall'applicazione nelle chiamate successive.
3. L'interceptor MSAL include il token di accesso nell'`Authorization`intestazione di una richiesta HTTPS per chiamare Blu Age Runtime. AWS
4. La `extension-oauth` libreria AWS Blu Age estrae le informazioni dell'utente dall'intestazione utilizzando un file di configurazione AWS Blu Age Runtime (`application-main.yml`) e inserisce queste informazioni in un `SharedContext` oggetto in modo che la logica aziendale possa utilizzarle.

   > **Note**
   >
   > `SharedContext`è un componente di runtime fornito da AWS Blu Age che gestisce il contesto e le informazioni sullo stato dell'applicazione modernizzata. Per ulteriori informazioni sui componenti e sugli aggiornamenti di AWS Blu Age Runtime, consulta le note di rilascio di AWS Blu Age nella Modernizzazione del mainframe AWS documentazione. Per ulteriori informazioni sul `application-main.yml` file, consulta Configurare la configurazione per AWS Blu Age Runtime nella Modernizzazione del mainframe AWS documentazione.

5. Il AWS Blu Age Runtime verifica se il token è presente.
   a. Se il token è presente, ne verifica la validità comunicando con Microsoft Entra ID.
   b. Se il token non è presente, AWS Blu Age Runtime restituisce un errore con il codice di stato HTTP 403.
6. Se il token è valido, AWS Blue Age Runtime consente alla logica aziendale di continuare. Se il token non è valido, AWS Blu Age Runtime restituisce un errore con il codice di stato HTTP 403.

OAuth Flusso di lavoro 2.0

Per un diagramma di alto livello del flusso di lavoro OAuth 2.0, consulta la documentazione di Microsoft Entra.

Tools (Strumenti)

Servizi AWS

Modernizzazione del mainframe AWSfornisce strumenti e risorse per aiutarvi a pianificare e implementare la migrazione e la modernizzazione dai mainframe agli AWS ambienti di runtime gestiti. È possibile utilizzare le funzionalità di refactoring di questo servizio, fornite da AWS Blu Age, per convertire e modernizzare le applicazioni mainframe esistenti.

> **Note**
>
> Modernizzazione del mainframe AWS Il servizio (Managed Runtime Environment experience) non è più aperto a nuovi clienti. Per funzionalità simili a Modernizzazione del mainframe AWS Service (Managed Runtime Environment experience), esplora Modernizzazione del mainframe AWS Service (Self-Managed Experience). I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta Modifica della Modernizzazione del mainframe AWS disponibilità.

Archivio di codice

L' CardDemo applicazione è stata aggiornata per dimostrare l'integrazione con Microsoft Entra ID. È possibile accedere al codice dal GitHub repository per questo pattern.

Configurazione del backend

Questo modello richiede modifiche al file di `application-main.yml` configurazione per abilitare Spring Security utilizzando OAuth 2.0 sull'applicazione di backend. Il `.yml` file ha il seguente aspetto:

```
gapwalk-application.security: enabled
gapwalk-application:
  security:
    identity: oauth
    issuerUri: ${issuerUrl}
    claim:
      claims:
        -
          claimName: upn
          claimMapValue: username
spring:
  autoconfigure:
    exclude:
     -
 org.springframework.boot.autoconfigure.security.oauth2.client.servlet.OAuth2ClientAutoConfigur
```

```
    -
 org.springframework.boot.autoconfigure.security.oauth2.resource.servlet.OAuth2ResourceServerAu
  security:
    oauth2:
      client:
        registration:
          azure:
            client-id: {clientId}
            client-secret: ${clientSecret}
            provider: azure
            authorization-grant-type: authorization_code
            redirect-uri: ${redirectUri}
            scope: openid

        provider:
          azure:
            authorization-uri: ${gapwalk-application.security.issuerUri}/oauth2/v2.0/
authorize
            token-uri:  ${gapwalk-application.security.issuerUri}/oauth2/v2.0/token
            jwk-set-uri: ${gapwalk-application.security.issuerUri}/discovery/v2.0/keys
      resourceserver:
        jwt:
          jwk-set-uri: ${gapwalk-application.security.issuerUri}/discovery/v2.0/keys
```

AWS Libreria di filtri di OAuth estensione Blu Age

La libreria di OAuth estensioni AWS Blu Age viene fornita dal team AWS Blu Age durante la collaborazione con AWS Professional Services.

Questa libreria legge la `claim.claims` configurazione nel `application-main.yml` file mostrata nel blocco di codice precedente. Questa configurazione è un elenco. Ogni elemento dell'elenco fornisce due valori: `claimName` e`claimMapValue`. `claimName`rappresenta un nome di chiave in un JSON Web Token (JWT) inviato dal frontend ed `claimMapValue` è il nome della chiave in ingresso. `SharedContext` Ad esempio, se desideri acquisire l'ID utente sul backend, imposta `claimName` il nome della chiave nel JWT che contiene l'ID `userId` that fornito da Microsoft Entra e imposta il nome della chiave `claimMapValue` per recuperare l'ID utente nel codice di backend.

Ad esempio, se si imposta `UserId``claimMapValue`, è possibile utilizzare il codice seguente per estrarre l'ID utente:

```
SharedContext.get().getValue("userId", [UserId]);
```

Best practice

Nell'implementazione di questo modello, tenete conto delle seguenti importanti considerazioni sulla sicurezza.

> **Important**
>
> Questo modello fornisce una base per l'integrazione dell'autenticazione. Si consiglia di implementare misure di sicurezza oltre a quelle illustrate in questa sezione in base ai requisiti aziendali prima di implementarlo in produzione.

- AWS sicurezza della configurazione.Sposta i valori di configurazione sensibili da `application-main.yml` a Gestione dei segreti AWS. Ad esempio, configura le seguenti proprietà utilizzando Secrets Manager:

```
security:
    oauth2:
      client:
        registration:
          azure:
            client-id: {clientId}
            client-secret: ${clientSecret}
```

  Per ulteriori informazioni su come utilizzare Secrets Manager per configurare i parametri AWS Blu Age, consulta AWS Blu Age Runtime secrets nella Modernizzazione del mainframe AWS documentazione.

- Protezione dell'ambiente di runtime. Configura l'ambiente applicativo modernizzato con controlli di AWS sicurezza adeguati:

```
server:
  tomcat:
    remoteip:
     protocol-header: X-Forwarded-Proto
     remote-ip-header: X-Forwarded-For
  forward-headers-strategy: NATIVE
```

- CloudWatch Registrazione su Amazon. Valuta la possibilità di aggiungere il file`logback-spring.xml to src/main/resources`:

```
<configuration>
 <appender name="CLOUDWATCH"
 class="com.amazonaws.services.logs.logback.CloudWatchAppender">
   <logGroup>/aws/bluage/application</logGroup>
   <logStream>${AWS_REGION}-${ENVIRONMENT}</logStream>
   <layout>
    <pattern>%d{yyyy-MM-dd HH:mm:ss} [%thread] %-5level %logger{36} - %msg%n</
pattern>
   </layout>
 </appender>

 <root level="INFO">
 <appender-ref ref="CLOUDWATCH"/>
 </root>
</configuration>
```

Per informazioni sull'abilitazione del tracciamento con CloudWatch, consulta Enable trace to log correlation nella CloudWatch documentazione.

- Configurazione e gestione dei token. Configura la durata dei token in Microsoft Entra ID per allinearla ai tuoi requisiti di sicurezza. Imposta la scadenza dei token di accesso entro 1 ora e aggiorna i token in modo che scadano entro 24 ore. Nella configurazione AWS Blu Age Runtime (`application-main.yml`), assicurati che la convalida JWT sia configurata correttamente con l'URI esatto dell'emittente e i valori di audience rilevati durante la registrazione dell'applicazione Entra ID.

  Quando un token scade e viene aggiornato:

  1. L'intercettore degli errori dell'applicazione Angular gestisce la risposta 401 ottenendo un nuovo token tramite MSAL.
  2. Il nuovo token viene inviato con la richiesta successiva.
  3. Il OAuth filtro di AWS Blu Age Runtime convalida il nuovo token e si aggiorna automaticamente `SharedContext` con le informazioni utente correnti. Ciò garantisce che la business logic continui ad avere accesso a un contesto utente valido tramite `SharedContext.get().getValue()` chiamate.

  Per ulteriori informazioni sui componenti AWS Blu Age Runtime e sui relativi aggiornamenti, consulta le note di rilascio di AWS Blu Age.

- Sicurezza di AWS Blu Age Runtime. La `oauth2-ext` libreria fornita da AWS Blu Age deve essere collocata nella posizione corretta della directory condivisa (`{app-server-home}/shared/`) con le autorizzazioni appropriate per i file. Verifica che la libreria estragga correttamente le informazioni dell'utente JWTs controllando la popolazione degli `SharedContext` oggetti nei tuoi log.
- Configurazione dei reclami specifici. Nel`application-main.yml`, definisci in modo esplicito le attestazioni di cui hai bisogno da Microsoft Entra ID. Ad esempio, per acquisire l'e-mail e i ruoli dell'utente, specifica:

```
gapwalk-application:
  security:
    claim:
      claims:
        - claimName: upn
          claimMapValue: username
        - claimName: roles
          claimMapValue: userRoles
        - claimName: email
          claimMapValue: userEmail
```

- Gestione degli errori. Aggiungi la gestione degli errori per risolvere gli errori di autenticazione nella tua applicazione Angular; ad esempio:

```
@Injectable()
export class AuthErrorInterceptor implements HttpInterceptor {
  intercept(request: HttpRequest<any>, next: HttpHandler): Observable<HttpEvent<any>>
 {
    return next.handle(request).pipe(
      catchError((error: HttpErrorResponse) => {
        if (error.status === 401) {
          // Handle token expiration
          this.authService.login();
        }
        if (error.status === 403) {
          // Handle unauthorized access
          this.router.navigate(['/unauthorized']);
        }
        return throwError(() => error);
      })
    );
  }
}
```

- Configurazione del timeout della sessione. Configura le impostazioni di timeout della sessione sia in AWS Blu Age Runtime che in Microsoft Entra ID. Ad esempio, aggiungi il codice seguente al tuo `application-main.yml` file:

```
server:
  servlet:
    session:
      timeout: 3600 # 1 hour in seconds
```

- MsalGuard. È necessario implementare la MsalGuard funzionalità per tutte le rotte protette per impedire l'accesso non autorizzato. Esempio:

```
const routes: Routes = [
    { path: '', redirectTo: '/transaction-runner', pathMatch: 'full' },
    { path: 'transaction-runner', component: TransactionRunnerComponent,
 canActivate:guards },
    { path: 'user-info', component: UserInfoComponent, canActivate:guards },
    { path: 'term/:transid/:commarea', component: TermComponent,
 canActivate:guards },
 { path: 'code', component: TransactionRunnerComponent  }
];
```

Le rotte prive di MsalGuard protezione saranno accessibili senza autenticazione, esponendo potenzialmente funzionalità sensibili. Assicurati che tutte le rotte che richiedono l'autenticazione includano le guardie nella loro configurazione.

Epiche

Configurare un ID Microsoft Entra

Operazione	Description	Competenze richieste
Configura un account Microsoft Azure per creare un ID Entra.	Per opzioni e istruzioni, consulta il sito Web di Microsoft Azure.	Sviluppatore di app
Configura un ID Microsoft Entra nella tua applicazione.	Per sapere come aggiunger e l'autenticazione Microsoft Entra ID B2C (Azure AD B2C)	Sviluppatore di app

Operazione	Description	Competenze richieste
	alla tua Angular SPA, consulta la documentazione Microsoft. Nello specifico: 1. Registra la tua applicazione e registra gli identificatori. 2. Esponi l'endpoint di transazione AWS Blu Age aggiungendo ed esponendo un ambito personalizzato. 3. Dopo aver definito l'ambito e l'URL dell'API, configura questi valori `apiUri` e le `apiScope` proprietà del `environment.ts` file nel repository di codice.	

Clona il repository e distribuisci il tuo codice Blu Age AWS

Operazione	Description	Competenze richieste
Clona il GitHub repository per ottenere il codice Angular richiesto per l'autenticazione.	Esegui il seguente comando per clonare il GitHub repository fornito con questo pattern nella tua directory di lavoro corrente locale: `git clone https://github.com/aws-samples/sample-microsoft-entra-id-based-auth-in-aws-bluage-modernized-mainframe-app.git`	Sviluppatore di app

Operazione	Description	Competenze richieste
Implementa il codice modernizzato AWS Blu Age su un server Tomcat per implementare l'autenticazione.	Per configurare l'ambiente locale che include Tomcat e il server di sviluppo Angular, segui i passaggi di installaz ione forniti dal team AWS Blu Age nell'ambito del coinvolgi mento dei clienti con Professio nal Services. AWS	Sviluppatore di app

Crea la soluzione di autenticazione

Operazione	Description	Competenze richieste
Abilita la sicurezza di AWS Blu Age Runtime per proteggere gli endpoint dell'API REST di AWS Blu Age.	Configura il `application-main.yml` file utilizzato da AWS Blu Age Runtime come segue. Per un esempio di questo file, consultate la sezione Code repository riportata in precedenza in questo modello. • `spring:security:oauth2:client:*` abilita il supporto client Spring Security OAuth 2.0. Sostituisci le impostazioni con valori specifici per la configurazione del tuo ID Microsoft Entra. • `spring:security:oauth2:resourceserver` abilita il supporto del	Sviluppatore di app

Operazione	Description	Competenze richieste
	server di risorse Spring Security OAuth 2.0. • `gapwalk-application.security.issuerUri` specifica l'URL del provider di identità per il recupero di informazioni di configurazione come l'endpoint di autorizzazione, l'endpoint del token e così via. • `gapwalk-application.security.identity` deve essere impostato su. `oauth` • `gapwalk-application.security.claim.claims` specifica l'elenco delle attestazioni richieste per acquisire utenti IDs e nomi utente. Per un esempio, consultate la precedente sezione Code repository in questo modello. • `gapwalk-application.security` deve essere impostato su. `enabled` • `spring.autoconfigure.exclude` deve essere rimosso per abilitare l'autenticazione.	

Operazione	Description	Competenze richieste
Incorpora il codice di esempio dal tuo ambiente locale nella tua base di codice Angular modernizzata di Blu Age.	Per informazioni su come incorporare l'esempio nella tua base di codice Angular modernizzata da AWS Blu Age, consulta la sezione Code repository precedente in questo modello.	Sviluppatore di app

Posiziona la `oauth2-ext` libreria nella directory condivisa.	Posiziona la **`oauth2-ext`** libreria nella directory condivisa del server delle applicazioni in modo che la tua applicazione modernizzata AWS Blu Age possa utilizzarla.Esegui i comandi seguenti:
	`cd oauth2-ext/target`
	`cp extension-oauth-filter-<version>.jar /{app-server-home}/shared/`
		Sviluppatore di app

Implementa la soluzione di autenticazione

Operazione	Description	Competenze richieste

Implementa l'applicazione frontend.	Esegui i seguenti comandi per avviare l'applicazione frontend localmente:
	`npm install`
	`ng serve --ssl`
	`npm start`
		Sviluppatore di app

Operazione	Description	Competenze richieste
	Note L'aggiunta del `--ssl` flag al `ng serve` comando garantisce che il server di sviluppo utilizzi HTTPS, che è più sicuro di altri protocolli e fornisce una migliore simulazione di un ambiente di produzione.	
Avvia l'applicazione di backend.	Avvia il server Tomcat in Eclipse.	Sviluppatore di app

Eseguire il test dell'applicazione

Operazione	Description	Competenze richieste
Verifica la funzionalità di accesso.	Accedi all'applicazione distribuita localmente all'indirizzo `http://localhost:4200` per verificare che agli utenti venga chiesto di confermare la propria identità. **Note** HTTP viene utilizzato qui a scopo dimostrativo. In un ambiente di produzione o in	Sviluppatore di app

Operazione	Description	Competenze richieste
	un altro ambiente accessibile al pubblico, è necessari o utilizzare HTTPS per motivi di sicurezza. Anche per lo sviluppo locale, ti consigliamo di configurare HTTPS quando possibile. Dovrebbe apparire la richiesta di accesso Microsoft e gli utenti configurati in Microsoft Entra ID dovrebbero poter accedere all'applicazione.	

Operazione	Description	Competenze richieste
Verifica l'intestazione di autorizzazione nella richiesta.	**Note** I passaggi seguenti utilizzano l'CardDemoapplicazione come esempio. Le fasi di test per altre applicazioni moderne possono variare. 1. Avvia la transazione `CC00` nell' CardDemo applicazione. 2. Accedi con le credenziali utente. Ad esempio, se stai utilizzando il workshop di autoapprendimento AWS Blu Age di livello 3, puoi utilizzare il nome `USER0001` e la password `PASSWORD` come credenziali. 3. Apri la finestra Strumenti per sviluppatori nella stessa scheda del browser. 4. Apri la scheda Rete e controlla la richiesta inviata al backend`http://localhost:8080/gapwalk-application/transaction` . 5. Controlla l'intestazione della richiesta e verifica di	Sviluppatore di app

Operazione	Description	Competenze richieste
	visualizzare un'intestazione di autorizzazione con il valore `Bearer <token>` where `<token>` è un valore generato da Microsoft Entra ID.	
Verifica la funzionalità di disconnessione.	Scegli Esci per disconnetterti e riprova ad accedere all'applicazione. Dovrebbe presentare una nuova richiesta di accesso.	Sviluppatore di app

risoluzione dei problemi

Problema	Soluzione
Il token emesso da Microsoft Entra ID non è compatibile con la sicurezza Spring Boot OAuth 2.0.	Per una risoluzione del problema, vedi Microsoft Entra ID OAuth Flow sul OAuth blog.
Domande generali relative ai token.	Per decodificare e visualizzare il contenuto di un token JWT, utilizza il sito Web https://jwt.io/.

Risorse correlate

- Per informazioni sul refactoring dell'applicazione utilizzando AWS Blu Age, consulta la documentazione.Modernizzazione del mainframe AWS
- Per capire come funziona la OAuth versione 2.0, visitate il sito Web OAuth 2.0.
- Per una panoramica della Microsoft Authentication Library (MSAL), consulta la documentazione di Microsoft Entra.
- Per informazioni sui profili utente su un sistema AS/400, consulta il tutorial IBM i (AS400).

- Per il flusso di autenticazione OAuth 2.0 e OpenID Connect (OIDC) nella piattaforma di identità Microsoft, consulta la documentazione di Microsoft Entra.

Integra il controller universale Stonebranch con la modernizzazione del mainframe AWS

Vaidy Sankaran e Pablo Alonso Prieto, Amazon Web Services

Robert Lemieux e Huseyin Gomleksizoglu, Stonebranch

Riepilogo

Nota: Modernizzazione del mainframe AWS Service (Managed Runtime Environment experience) non è più aperto a nuovi clienti. Per funzionalità simili a Modernizzazione del mainframe AWS Service (Managed Runtime Environment experience), esplora Modernizzazione del mainframe AWS Service (Self-Managed Experience). I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta Modifica della Modernizzazione del mainframe AWS disponibilità.

Questo modello spiega come integrare l'orchestrazione del carico di lavoro Stonebranch Universal Automation Center (UAC) con il servizio di modernizzazione del mainframe di Amazon Web Services (AWS). Il servizio AWS Mainframe Modernization migra e modernizza le applicazioni mainframe nel cloud AWS. Offre due modelli: AWS Mainframe Modernization Replatform con tecnologia Micro Focus Enterprise e AWS Mainframe Modernization Automated Refactor con AWS Blu Age.

Stonebranch UAC è una piattaforma di automazione e orchestrazione IT in tempo reale. L'UAC è progettato per automatizzare e orchestrare lavori, attività e flussi di lavoro su sistemi IT ibridi, da quelli locali a AWS. I clienti aziendali che utilizzano sistemi mainframe stanno passando a infrastrutture e applicazioni modernizzate incentrate sul cloud. Gli strumenti e i servizi professionali di Stonebranch facilitano la migrazione degli scheduler e delle funzionalità di automazione esistenti nel cloud AWS.

Quando migri o modernizzi i tuoi programmi mainframe sul cloud AWS utilizzando il servizio AWS Mainframe Modernization, puoi utilizzare questa integrazione per automatizzare la pianificazione in batch, aumentare l'agilità, migliorare la manutenzione e ridurre i costi.

Questo modello fornisce istruzioni per l'integrazione dello scheduler Stonebranch con le applicazioni mainframe migrate al runtime Micro Focus Enterprise del servizio AWS Mainframe Modernization. Questo modello è destinato agli architetti di soluzioni, agli sviluppatori, ai consulenti, agli specialisti della migrazione e a tutti coloro che si occupano di migrazioni, modernizzazioni, operazioni o. DevOps

Risultato mirato

Questo modello si concentra sul raggiungimento dei seguenti obiettivi:

- La capacità di pianificare, automatizzare ed eseguire processi batch mainframe in esecuzione nel servizio AWS Mainframe Modernization (runtime Microfocus) di Stonebranch Universal Controller.
- Monitora i processi batch dell'applicazione dallo Stonebranch Universal Controller.
- Start/Restart/Rerun/Stopi processi in batch vengono eseguiti automaticamente o manualmente dallo Stonebranch Universal Controller.
- Recupera i risultati dei processi batch di AWS Mainframe Modernization.
- Acquisisci i CloudWatch log AWS dei lavori in batch in Stonebranch Universal Controller.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un'applicazione Micro Focus Bankdemo con file Job Control Language (JCL) e un processo batch distribuito in un ambiente del servizio AWS Mainframe Modernization (Micro Focus runtime)
- Conoscenza di base su come creare e distribuire un'applicazione mainframe eseguita su Micro Focus Enterprise Server
- Conoscenza di base di Stonebranch Universal Controller
- Licenza di prova di Stonebranch (contatta Stonebranch)
- Istanze Amazon Elastic Compute Cloud (Amazon EC2) per Windows o Linux (ad esempio, xlarge) con un minimo di quattro core, 8 GB di memoria e 2 GB di spazio su disco
- Apache Tomcat versione 8.5.x o 9.0.x
- Oracle Java Runtime Environment (JRE) o OpenJDK versione 8 o 11
- Edizione compatibile con Amazon Aurora MySQL
- Bucket Amazon Simple Storage Service (Amazon S3) per repository di esportazione
- Amazon Elastic File System (Amazon EFS) per le connessioni degli agenti Stonebranch Universal Message Service (OMS) per l'alta disponibilità (HA)
- File di installazione di Stonebranch Universal Controller 7.2 Universal Agent 7.2
- Modello di pianificazione delle attività di AWS Mainframe Modernization (ultima versione rilasciata del file.zip)

Limitazioni

- Il prodotto e la soluzione sono stati testati e la compatibilità è stata convalidata solo con OpenJDK 8 e 11.
- Il modello aws-mainframe-modernization-stonebranchdi pianificazione delle attività di integrazione funzionerà solo con il servizio AWS Mainframe Modernization.
- Questo modello di pianificazione delle attività funzionerà solo su un'edizione Unix, Linux o Windows degli agenti Stonebranch.
- Alcuni servizi AWS non sono disponibili in tutte le regioni AWS. Per la disponibilità regionale, consulta i servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Architecture

Architettura dello stato di destinazione

Il diagramma seguente mostra l'esempio di ambiente AWS richiesto per questo programma pilota.

1. Stonebranch Universal Automation Center (UAC) include due componenti principali: Universal Controller e Universal Agents. Stonebranch OMS viene utilizzato come bus di messaggi tra il controller e i singoli agenti.
2. Il database Stonebranch UAC viene utilizzato da Universal Controller. Il database può essere compatibile con MySQL, Microsoft SQL Server, Oracle o Aurora MySQL.
3. Servizio AWS Mainframe Modernization: ambiente di runtime Micro Focus con l'BankDemo applicazione distribuita. I file BankDemo dell'applicazione verranno archiviati in un bucket S3. Questo bucket contiene anche i file JCL del mainframe.
4. Stonebranch UAC può eseguire le seguenti funzioni per l'esecuzione in batch:
    a. Avvia un processo batch utilizzando il nome del file JCL presente nel bucket S3 collegato al servizio di modernizzazione del mainframe AWS.
    b. Ottieni lo stato dell'esecuzione del processo batch.
    c. Attendi il completamento dell'esecuzione del processo batch.
    d. Recupera i registri dell'esecuzione del processo batch.
    e. Esegui nuovamente i processi batch non riusciti.
    f. Annulla il processo batch mentre il processo è in esecuzione.

5. Stonebranch UAC può eseguire le seguenti funzioni dell'applicazione:
    a. Avvia l'applicazione
    b. Ottieni lo stato della domanda
    c. Attendi che l'applicazione venga avviata o interrotta
    d. Interrompi l'applicazione
    e. Recupera i registri delle operazioni dell'applicazione

Conversione dei lavori di Stonebranch

Il diagramma seguente rappresenta il processo di conversione del lavoro di Stonebranch durante il percorso di modernizzazione. Descrive come le pianificazioni dei lavori e le definizioni delle attività vengono convertite in un formato compatibile in grado di eseguire attività batch di AWS Mainframe Modernization.

1. Per il processo di conversione, le definizioni dei job vengono esportate dal sistema mainframe esistente.
2. I file JCL possono essere caricati nel bucket S3 per l'applicazione Mainframe Modernization in modo che questi file JCL possano essere distribuiti dal servizio AWS Mainframe Modernization.
3. Lo strumento di conversione converte le definizioni di lavoro esportate in attività UAC.
4. Dopo aver creato tutte le definizioni delle attività e le pianificazioni dei lavori, questi oggetti verranno importati nell'Universal Controller. Le attività convertite eseguono quindi i processi nel servizio AWS Mainframe Modernization anziché eseguirli sul mainframe.

Architettura Stonebranch UAC

Il seguente diagramma di architettura rappresenta un active-active-passive modello di controller universale ad alta disponibilità (HA). Stonebranch UAC è distribuito in più zone di disponibilità per fornire alta disponibilità e supportare il disaster recovery (DR).

Controller universale

Due server Linux vengono forniti come controller universali. Entrambi si connettono allo stesso endpoint del database. Ogni server ospita un'applicazione Universal Controller e un OMS. La versione più recente di Universal Controller viene utilizzata al momento del provisioning.

Gli Universal Controller vengono distribuiti nella webapp Tomcat come documento ROOT e vengono serviti sulla porta 80. Questa implementazione semplifica la configurazione del load balancer del frontend.

HTTP over TLS o HTTPS è abilitato utilizzando il certificato wildcard di Stonebranch (ad esempio,). `https://customer.stonebranch.cloud` Ciò protegge la comunicazione tra il browser e l'applicazione.

OMS

Un Universal Agent e un OMS (Opswise Message Service) risiedono su ogni server Universal Controller. Tutti gli Universal Agent distribuiti dal cliente sono configurati per connettersi a entrambi i servizi OMS. OMS funge da servizio di messaggistica comune tra gli Universal Agent e l'Universal Controller.

Amazon EFS monta una directory di spool su ogni server. OMS utilizza questa directory di spool condivisa per mantenere le informazioni sulla connessione e sulle attività lontane da controller e agenti. OMS funziona in modalità ad alta disponibilità. Se l'OMS attivo non funziona, l'OMS passivo ha accesso a tutti i dati e riprende automaticamente le operazioni attive. Gli Universal Agent rilevano questa modifica e si connettono automaticamente al nuovo OMS attivo.

Database

Amazon Relational Database Service (Amazon RDS) ospita il database UAC, con Amazon Aurora MySQL come motore. Amazon RDS aiuta a gestire e offrire backup pianificati a intervalli regolari. Entrambe le istanze Universal Controller si connettono allo stesso endpoint del database.

Sistema di bilanciamento del carico

Per ogni istanza è configurato un Application Load Balancer. Il load balancer indirizza il traffico verso il controller attivo in qualsiasi momento. I nomi di dominio delle istanze puntano ai rispettivi endpoint di bilanciamento del carico.

URLs

Ciascuna delle tue istanze ha un URL, come illustrato nell'esempio seguente.

Ambiente	Istanza
Produzione	`customer.stonebranch.cloud`

Sviluppo (non produzione)	`customerdev.stonebranch.cloud`
Test (non di produzione)	`customertest.stonebranch.cloud`

> **Note**
>
> I nomi delle istanze non di produzione possono essere impostati in base alle proprie esigenze.

Alta disponibilità

L'alta disponibilità (HA) è la capacità di un sistema di funzionare ininterrottamente senza guasti per un determinato periodo di tempo. Tali errori includono, a titolo esemplificativo, lo storage, i ritardi di risposta nelle comunicazioni con il server causati da problemi di CPU o memoria e la connettività di rete.

Per soddisfare i requisiti HA:

- Tutte le EC2 istanze, i database e le altre configurazioni vengono replicati su due zone di disponibilità separate all'interno della stessa regione AWS.
- Il controller viene fornito tramite un'Amazon Machine Image (AMI) su due server Linux nelle due zone di disponibilità. Ad esempio, se hai effettuato il provisioning nella regione Europa eu-west-1, hai un controller universale nella zona di disponibilità eu-west-1a e nella zona di disponibilità eu-west-1c.
- Nessun processo può essere eseguito direttamente sui server delle applicazioni e nessun dato può essere archiviato su questi server.
- L'Application Load Balancer esegue controlli di integrità su ogni Universal Controller per identificare quello attivo e indirizzare il traffico verso di esso. Nel caso in cui un server presenti problemi, il load balancer porta automaticamente l'Universal Controller passivo a uno stato attivo. Il load balancer identifica quindi la nuova istanza attiva di Universal Controller dai controlli di integrità e inizia a indirizzare il traffico. Il failover avviene entro quattro minuti senza perdita di lavoro e l'URL del frontend rimane lo stesso.
- Il servizio di database compatibile con Aurora MySQL archivia i dati di Universal Controller. Per gli ambienti di produzione, un cluster di database è creato con due istanze di database in due diverse zone di disponibilità all'interno di una singola regione AWS. Entrambi gli Universal Controller

utilizzano un'interfaccia Java Database Connectivity (JDBC) che punta a un singolo endpoint del cluster di database. Nel caso in cui un'istanza di database presenti problemi, l'endpoint del cluster di database punta dinamicamente all'istanza integra. Non sono richiesti interventi manuali.

Backup ed eliminazione

Stonebranch Universal Controller è configurato per eseguire il backup e l'eliminazione dei vecchi dati seguendo la pianificazione mostrata nella tabella.

Tipo	Schedule
Attività	7 giorni
Audit	90 giorni
Cronologia	60 giorni

I dati di backup precedenti alle date mostrate vengono esportati in formato.xml e archiviati nel file system. Una volta completato il processo di backup, i dati più vecchi vengono eliminati dal database e archiviati in un bucket S3 per un massimo di un anno per le istanze di produzione.

Puoi modificare questa pianificazione nell'interfaccia Universal Controller. Tuttavia, l'aumento di questi intervalli di tempo potrebbe causare tempi di inattività più lunghi durante la manutenzione.

Tools (Strumenti)

Servizi AWS

- AWS Mainframe Modernization è una piattaforma AWS nativa per il cloud che ti aiuta a modernizzare le tue applicazioni mainframe in ambienti di runtime gestiti da AWS. Offre strumenti e risorse per aiutarti a pianificare e implementare la migrazione e la modernizzazione.
- Amazon Elastic Block Store (Amazon EBS) fornisce volumi di storage a livello di blocco da utilizzare con le istanze Amazon. EC2
- Amazon Elastic File System (Amazon EFS) ti aiuta a creare e configurare file system condivisi nel cloud AWS.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS. Questo modello utilizza l'edizione compatibile con Amazon Aurora MySQL.

- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Elastic Load Balancing (ELB) distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni. Ad esempio, puoi distribuire il traffico tra EC2 istanze Amazon, contenitori e indirizzi IP in una o più zone di disponibilità. Questo modello utilizza un Application Load Balancer.

Stonebranch

- Universal Automation Center (UAC) è un sistema di prodotti per l'automazione dei carichi di lavoro aziendali. Questo modello utilizza i seguenti componenti UAC:
  - Universal Controller, un'applicazione Web Java in esecuzione in un contenitore Web Tomcat, è la soluzione aziendale per la pianificazione dei lavori e l'automazione del carico di lavoro di Universal Automation Center. Il controller presenta un'interfaccia utente per la creazione, il monitoraggio e la configurazione delle informazioni sul controller, gestisce la logica di pianificazione, elabora tutti i messaggi da e verso gli Universal Agent e sincronizza gran parte delle operazioni ad alta disponibilità di Universal Automation Center.
  - Universal Agent è un agente di pianificazione indipendente dal fornitore che collabora con Job Scheduler esistente su tutte le principali piattaforme informatiche, sia legacy che distribuite. Sono supportati tutti gli scheduler che funzionano su Unixz/Series, i/Series, Linux o Windows.
- Universal Agent è un agente di pianificazione indipendente dal fornitore che collabora con Job Scheduler esistente su tutte le principali piattaforme informatiche, sia legacy che distribuite. Sono supportati tutti gli scheduler che funzionano su Unixz/Series, i/Series, Linux o Windows.
- Stonebranch -integration aws-mainframe-modernization-stonebranch AWS Mainframe Modernization Universal Extension è il modello di integrazione per eseguire, monitorare e rieseguire lavori in batch nella piattaforma AWS Mainframe Modernization.

Codice

Il codice per questo pattern è disponibile nel repository -integration. aws-mainframe-modernization-stonebranch GitHub

Epiche

Installa Universal Controller e Universal Agent su Amazon EC2

Operazione	Description	Competenze richieste
Scarica i file di installazione.	Scarica l'installazione dai server Stonebranch. Per ottenere i file di installazione, contatta Stonebranch.	Architetto del cloud
Avvia l' EC2 istanza.	Avrai bisogno di circa 3 GB di spazio aggiuntivo per le installazioni di Universal Controller e Universal Agent. Fornisci quindi almeno 30 GB di spazio su disco per l'istanza . Aggiungi la porta 8080 al gruppo di sicurezza in modo che sia accessibile.	Architetto del cloud

Verifica i prerequisiti.	Prima dell'installazione, procedi come segue:
	
	1. Installate Java come descritto in Downloading Java Runtime Environment.
	
	``
	$ sudo yum -y update
	$ sudo yum install
	 java-11-amazon-cor
	retto
	``
	
	Assicuratevi di utilizzare una delle versioni JAVA supportate. Il comando precedente dovrebbe
		Amministratore cloud, amministratore Linux

Operazione	Description	Competenze richieste

	installare java-11. Controlla la versione Java e assicurati di utilizzare la versione 11 prima di continuare.
	2. Come descritto in Installazione di Apache Tomcat, esegui i seguenti comandi.
	`$ sudo yum install tomcat tomcat-admin-webapps`
	`$ sudo systemctl enable tomcat`
	`$ sudo systemctl start tomcat`
	3. Crea un database Amazon Aurora come descritto in Creazione e connessione a un cluster Aurora MySQL DB. Usa l'edizione compatibile con Amazon Aurora MySQL.
	Scegli un nome utente e una password principali. Mantieni i valori predefiniti per il resto delle impostazioni.	

Operazione	Description	Competenze richieste

Installa Universal Controller.	1. Carica il file `universal-controller-7.2.0.0.tar` di installazione sull'EC2 istanza.
	2. Estrai l'archiviazione dei file di installazione in una `temp` cartella.
	`$ tar -xvf universal-controller-7.2.0.0.tar`
	3. Concedi l'autorizzazione all'esecuzione dello script di installazione.
	`$ chmod a+x install-controller.sh`
	4. Installare il controller. Questo esempio utilizza il comando seguente per installare Universal Controller sotto `/usr/share/tomcat`. Usa il database Amazon Aurora creato nei passaggi precedenti.
	`$ sudo ./install-controller.sh --tomcat-dir /usr/share/tomcat/ --controller-file universal-controller-7.2.0.0-build.145.war --dbuser admin --dbpass`
		Architetto cloud, amministratore Linux

Operazione	Description	Competenze richieste
	"****" --dbname uc --rdbms mysql --dburl jdbc:mysql://database-2-instance-1.cih63miincgy.us-east-1.rds.amazonaws.com:3306/ L'ultima riga dell'output dello script dovrebbe essere «Installazione completa». 5. Vai al seguente URL nell' EC2 istanza. http://<public_ip>:8080/uc 6. Nella schermata di accesso, inserisci ops.admin nella sezione Nome utente e mantieni vuoto il campo Password. 7. Imposta una nuova password per l'ops.admin utente.	

Operazione	Description	Competenze richieste

Installa Universal Agent.	1. Carica il file `sb-7.2.0.1-linux-3.10-x86_64.tar.Z` di installazione sull' EC2 istanza.
	2. Accedi all' EC2 istanza.
	3. Annulla l'archiviazione del pacchetto di installazione di Universal Agent.
	`$ zcat sb-7.2.0.1-linux-3.10-x86_64.tar.Z | tar xvf -`
	4. Eseguire il seguente comando seguente.
	`$ sudo ./unvinst --oms_servers 7878@localhost --oms_autostart yes --python yes`
	5. Crea un file PAM.
	`$ cp /etc/pam.d/login /etc/pam.d/ucmd`
	6. Abilita l'avvio automatico per Universal Agent.
	`$ /sbin/restorecon -v /etc/rc.d/init.d/ubrokerd`
		Amministratore cloud, amministratore Linux

Operazione	Description	Competenze richieste
Aggiungi OMS a Universal Controller.	1. Accedi a Universal Controller con l'`ops.admin` utente. 2. Scegli il menu Servizi nell'angolo in alto a sinistra dello schermo, quindi scegli il menu OMS Servers nel Sistema 3. Nel campo Indirizzo server OMS`localhost` , digita e quindi salva. 4. Verrà visualizzato lo stato del server OMS come Connesso e lo stato della sessione come Operativo.	Amministratore di Universal Controller

Importa AWS Mainframe Modernization Universal Extension e crea un'attività

Operazione	Description	Competenze richieste
Importa modello di integrazione.	Per questo passaggio, è necessaria l'estensione universale di AWS Mainframe Modernization. Assicurati che sia stata scaricata l'ultima versione rilasciata del file.zip. 1. Accedi all'Universal Controller con l'`ops.admin` utente. 2. Vai a Servizi, Importa modello di integrazione.	Amministratore di Universal Controller

Operazione	Description	Competenze richieste
	3. Seleziona il file.zip del modello di integrazione (`aws_mainframe_modernization_stonebranch_extension.zip`) e scegli Importa. Dopo l'importazione del modello di integrazione, vedrai AWS Mainframe Modernization Tasks in Servizi disponibili.	

Operazione	Description	Competenze richieste
Abilita credenziali risolvibili.	1. Passa a Services, AWS Mainframe Modernization Tasks. 2. Nel pannello a destra, compila i campi obbligatori: • Nome: Nuova attività di modernizzazione del mainframe • Agente: seleziona re l'unico agente (AGNT0001). Nella sezione Dettagli sulla modernizzazione dei mainframe di AWS: • Azione: elenca gli ambienti • Credenziali AWS: se hai aggiunto un ruolo AWS Identity and Access Management (IAM) all' EC2 istanza, puoi lasciare questo campo vuoto. Se intendi utilizzar e `AWSAccessKeyID` eAWSSecretKey , scegli l'icona () accanto al campo. Nella finestra Dettagli delle credenziali che si apre, inserisci le seguenti informazioni e quindi salva.	Amministratore di Universal Controller

Operazione	Description	Competenze richieste
	• Nome: credenziali di modernizzazione dei mainframe AWS • Utente runtime: scrivi l'ID della chiave di accesso AWS in questo campo. • Password di runtime: scrivi la chiave segreta AWS in questo campo. • Punto finale: assicurati che l'endpoint abbia la regione AWS corretta. Il valore predefinito è `https://m2.us-east-1.amazonaws.com` . • Regione: entra nella regione del servizio AWS Mainframe Modernization. Il valore predefinito è `us-east-1` . 3. Mantieni i valori predefiniti nel resto dei campi e salva l'attività.	

Operazione	Description	Competenze richieste
Avvia l'operazione.	1. Nella parte superiore del pannello di destra, scegli Avvia attività. 2. Nella finestra di conferma, scegli Avvia. Dopodiché , la Universal Controlle r Console visualizzerà un messaggio simile al seguente messaggio. 2022-08-24 10:11:49 Lanciato con successo il task universale «New Mainframe Moderniza tion Task» con l'istanza di attività sys_id 166129149 3634146313 E38 OZJY. NC8 DB8 3. Vai alla scheda Istanze Se non vedi la scheda Istanze, scegli la freccia destra per scorrere verso destra. 4. Apri il menu contestuale (fai clic con il pulsante destro del mouse) per l'istanza dell'attività nell'elenco, scegli Recupera output, quindi scegli Invia in Recupera output 5. Nella finestra Recupera output, vedrai l'elenco degli ambienti in STDOUT.	Amministratore di Universal Controller

Test di avvio di un processo in batch

Operazione	Description	Competenze richieste
Crea un'attività per il processo batch.	1. Passa a Services, AWS Mainframe Modernization Tasks. 2. Nel pannello a destra, compila i campi obbligatori: • Nome: Nuova attività di modernizzazione del mainframe • Agente: seleziona re l'unico agente (AGNT0001). Nella sezione Dettagli sulla modernizzazione dei mainframe di AWS: • Azione: Avvia Batch (o Avvia Batch e attendi l'esecuzione del processo batch e attendi il completamento dell'atti vità in AWS) • Credenziali AWS: se hai aggiunto un ruolo IAM all' EC2 istanza, puoi lasciare questo campo vuoto. Se intendi utilizzar e `AWSAccessKeyID` e`AWSSecretKey` , scegli l'icona () accanto al campo. • Punto finale: assicurat i che l'endpoint abbia la	Amministratore di Universal Controller

Operazione	Description	Competenze richieste
	regione AWS corretta. Il valore predefinito è https://m2.us-east-1.amazonaws.com. • Regione: entra nella regione del servizio AWS Mainframe Modernization. Il valore predefinito è `us-east-1` . • Applicazione: scegli l'icona accanto al campo () e scegli Invia nelle scelte di aggiornamento dell'applicazione. Questo si connetterà al servizio AWS Mainframe Modernization e restituirà l'elenco delle applicazioni. Ora puoi selezionare l'applicazione dall'elenco a discesa. Seleziona l'applicazione su cui desideri eseguire il processo batch. • Nome file JCL: `RUNHELLO.jcl` • Attendi esito positivo o negativo: se questa opzione è selezionata, l'operazione attenderà che lo stato del processo batch sia riuscito o meno. • Intervallo di polling: è il periodo di tempo	

Operazione	Description	Competenze richieste
	che intercorre tra un sondaggio e l'altro. • Recupera registri di esecuzione: se seleziona ta, i registri verranno recuperati automatic amente al termine del processo batch. • Formato di registro: questo è il formato dei log da stampare. Può essere in formato testo o JSON. 3. Mantieni i valori predefiniti nel resto dei campi e salva l'attività.	

Operazione	Description	Competenze richieste
Avvia l'operazione.	1. Nella parte superiore del pannello di destra, scegli Avvia attività. 2. Nella finestra di conferma, scegli Avvia. Dopodiché , la Universal Controlle r Console visualizzerà un messaggio simile al seguente messaggio. 2022-08-24 11:11:59 Lanciato con successo l'attività universale «Mainframe Modernization Start Batch» con l'istanza di attività sys_id. <sys id> 3. Vai alla pagina Istanze Se non vedi la scheda Istanze, scegli la freccia destra per scorrere verso destra. 4. Apri il menu contestuale (fai clic con il pulsante destro del mouse) per l'istanza dell'attività nell'elenco, scegli Recupera output, quindi scegli Invia in Recupera output 5. Nella finestra Recupera output, vedrai l'elenco degli ambienti in STDOUT.	Amministratore di Universal Controller

Crea un flusso di lavoro per più attività

Operazione	Description	Competenze richieste
Copia le attività.	1. Apri il menu contestuale (clic con il pulsante destro del mouse) per l'attività di cui desideri creare copie e scegli Copia. 2. Nella finestra Copy AWS Mainframe Moderniza tion Task inserisci il seguente nuovo nome per la nuova attività: Mainframe Modernization Start Batch -. RUNAWS2 3. Copiate nuovamente l'operazione, utilizzando il seguente nome: Mainframe Modernization Start Batch -. RUNAWS3 4. Copiare nuovamente con l'operazione, utilizzando il seguente nome: Mainframe Modernization Start Batch -. RUNAWS4 5. Copiate l'operazione un'ultima volta, utilizzando il seguente nome: Mainframe Modernization Start Batch - FOOBAR.	Amministratore Universal Controller
Attività di aggiornamento.	1. Aprite (doppio clic) l' RUNAWS2 operazione Mainframe Modernization Start Batch -, modificate il	Amministratore di Universal Controller

Operazione	Description	Competenze richieste
	campo JCL File Name in e salvate`RUNAWS2.jcl` . 2. Aprite (doppio clic) l' RUNAWS3 operazione Mainframe Modernization Start Batch -, modificate il campo JCL File Name in e salvate*`RUNAWS3.jcl`* . 3. Aprite (doppio clic) l' RUNAWS4 operazione Mainframe Modernization Start Batch -, modificate il campo JCL File Name in e salvate*`RUNAWS4.jcl`* . 4. Aprire (doppio clic) l'attivit à Mainframe Moderniza tion Start Batch - FOOBAR, modificare il campo JCL File Name in e salvare. *`MISSING.jcl`* Questa operazione avrà esito negativo perché il valore del nome del file JCL non è corretto.	

Operazione	Description	Competenze richieste
Crea un flusso di lavoro.	1. Vai a Servizi, Flussi di lavoro. 2. Nel pannello di destra, inserisci Mainframe Modernization Workflow nel campo Nome e salva. 3. Nel pannello di destra, scegli Modifica flusso di lavoro. 4. Nella scheda Workflow Editor, il pulsante Aggiungi attività (+). 5. Nella finestra Task Find, scegli Cerca per visualizz are tutte le attività in Universal Controller. 6. Fate clic sull'icona accanto a Mainframe Modernization Start Batch Task e trascinat e l'icona in una posizione vuota dell'Editor di flussi di lavoro. 7. Ripetete la stessa azione per le altre attività di modernizzazione del mainframe e inseritel e come mostrato nella sezione Informazioni aggiuntive. 8. Scegli il pulsante Connect () e collega le attività tra loro. Per collegare un'attività a un'altra, fai clic al centro	Amministratore di Universal Controller

Operazione	Description	Competenze richieste
	di un'attività e trascinala sull'attività di destinazione. 9. Connect le attività come mostrato nella sezione Informazioni aggiuntive e salva il flusso di lavoro. 10. Fate clic con il pulsante destro del mouse su un punto vuoto nell'Editor di flussi di lavoro, scegliete Avvia flusso di lavoro, quindi scegliete OK.	

Operazione	Description	Competenze richieste
Controlla lo stato del flusso di lavoro.	1. Nel menu a sinistra, scegli l'Attività 2. Al centro della finestra, scegli Avvia. Nell'elenco verrà visualizzato l'elenco delle istanze di attività. 3. Apri (fai doppio clic) Mainframe Modernization Workflow nell'elenco oppure apri il menu contestuale (fai clic con il pulsante destro del mouse) e scegli Workflow Task Commands, View Workflow. Le attività verranno visualizzate come mostrato nella sezione Informazioni aggiuntive. La seconda attività avrebbe dovuto fallire perché hai utilizzato un file JCL mancante.	Amministratore di Universal Controller

Risolvi i problemi relativi ai processi batch non riusciti e rieseguili

Operazione	Description	Competenze richieste
Correggi l'operazione non riuscita e riesegui.	1. Aprite (fate doppio clic) sull'operazione non riuscita per visualizzare l'errore relativo all'operazione.	Amministratore di Universal Controller

Operazione	Description	Competenze richieste
	2. Sono disponibili due opzioni per correggere l'operazione non riuscita. • Correggi il nome del file JCL e impostalo su. `F00BAR.jcl` • Aggiungi il nome file JCL corretto al nome file JCL (Temp). Questo campo sovrascriverà il campo Nome file JCL. Per questo programma pilota, scegliete la seconda opzione e salvate l'istanza dell'operazione. 3. In Workflow Monitor, aprite il menu contestuale (con il pulsante destro del mouse) relativo all'opera zione non riuscita e scegliete Comandi, Esegui nuovamente. 4. Dopodiché, tutte le attività verranno completate correttamente.	

Crea le attività di avvio dell'applicazione e interrompi l'applicazione

Operazione	Description	Competenze richieste
Crea l'azione Avvia applicazi one.	1. Passa a Services, AWS Mainframe Modernization Tasks.	Amministratore di Universal Controller

Operazione	Description	Competenze richieste
	2. Nel pannello di destra, compila i campi obbligatori. • Nome: Applicazione Mainframe Modernization Start • Agente: seleziona re l'unico agente (AGNT0001) Nella sezione Dettagli sulla modernizzazione dei mainframe di AWS: • Azione: Avvia l'applicazi one • Credenziali AWS: se hai aggiunto un ruolo IAM all' EC2 istanza, puoi lasciare questo campo vuoto. Se intendi utilizzar e `AWSAccessKeyID` eAWSSecretKey , seleziona la credenzia le che hai creato in precedenza. • Punto finale: assicurat i che l'endpoint abbia la regione corretta. Il valore predefinito è `https://m2.us-east -1.amazonaws.com` . • Regione: entra nella regione del servizio AWS Mainframe Moderniza	

Operazione	Description	Competenze richieste
	tion. Il valore predefinito è `us-east-1` . • Applicazione: scegli l'icona accanto al campo () e scegli Invia nelle scelte di aggiornam ento dell'applicazione. Questo si connetterà al servizio AWS Mainframe Modernization e restituir à l'elenco delle applicazi oni. Ora puoi seleziona re l'applicazione dall'elen co a discesa. Seleziona l'applicazione su cui desideri eseguire il processo batch. • Attendi esito positivo o negativo: se questa opzione è selezionata, l'operazione attenderà che lo stato del processo batch sia riuscito o meno. • Intervallo di polling: è il periodo di tempo che intercorre tra un sondaggio e l'altro. • Recupera registri di esecuzione: se seleziona ta, i registri verranno recuperati automatic amente al termine del processo batch.	

Operazione	Description	Competenze richieste
	• Formato di registro: questo è il formato dei log da stampare. Può essere in formato testo o JSON. 3. Mantieni i valori predefiniti nel resto dei campi e salva l'attività. 4. Ora copia questa attività e crea un'attività per Stop Application. Cambia il nome in Mainframe Moderniz ation Stop Application e modifica l'azione in Stop Application.	

Creare un'attività Annulla esecuzione Batch

Operazione	Description	Competenze richieste
Creare l'azione Annulla Batch.	1. Passa a Services, AWS Mainframe Modernization Tasks. 2. Nel pannello di destra, compila i campi obbligatori. • Nome: Modernizzazione del mainframe Annulla l'esecuzione in batch • Agente: seleziona re l'unico agente (AGNT0001)	

Operazione	Description	Competenze richieste
	Nella sezione Dettagli sulla modernizzazione dei mainframe di AWS: • Azione: Annulla l'esecuzione del Batch • Credenziali AWS: se hai aggiunto un ruolo IAM all' EC2 istanza, puoi lasciare questo campo vuoto. Se intendi utilizzare `AWSAccessKeyID` e`AWSSecretKey` , seleziona la credenziale che hai creato in precedenza. • Punto finale: assicurati che l'endpoint abbia la regione corretta. Il valore predefinito è `https://m2.us-east-1.amazonaws.com` . • Regione: entra nella regione del servizio AWS Mainframe Modernization. Il valore predefinito è `us-east-1` . • Applicazione: scegli l'icona accanto al campo () e scegli Invia nelle scelte di aggiornamento dell'applicazione. Questo si connetterà al servizio AWS Mainframe	

Operazione	Description	Competenze richieste
	Modernization e restituir à l'elenco delle applicazi oni. Ora puoi seleziona re l'applicazione dall'elen co a discesa. Seleziona l'applicazione su cui desideri eseguire il processo batch. • Attendi esito positivo o negativo: se questa opzione è selezionata, l'operazione attenderà che lo stato del processo batch sia riuscito o meno. • Intervallo di polling: è il periodo di tempo che intercorre tra un sondaggio e l'altro. • Recupera registri di esecuzione: se seleziona ta, i registri verranno recuperati automatic amente al termine del processo batch. • Formato di registro: questo è il formato dei log da stampare. Può essere in formato testo o JSON. 3. Mantieni i valori predefiniti nel resto dei campi e salva l'attività.	

Risorse correlate

- Controller universale
- Agente universale
- Impostazioni LDAP
- Single Sign-On SAML
- Strumento di conversione Xpress

Informazioni aggiuntive

Icone nell'editor del flusso di lavoro

Tutte le attività connesse

Stato del flusso di lavoro

Esegui la migrazione e la replica di file VSAM su Amazon RDS o Amazon MSK utilizzando Connect from Precisely

Prachi Khanna e Boopathy GOPALSAMY, Amazon Web Services

Riepilogo

Questo modello mostra come migrare e replicare i file VSAM (Virtual Storage Access Method) da un mainframe a un ambiente di destinazione nel cloud AWS utilizzando Connect from Precisely. Gli ambienti di destinazione coperti da questo modello includono Amazon Relational Database Service (Amazon RDS) e Amazon Managed Streaming for Apache Kafka (Amazon MSK). Connect utilizza Change Data Capture (CDC) per monitorare continuamente gli aggiornamenti dei file VSAM di origine e quindi trasferirli in uno o più ambienti di destinazione AWS. Puoi utilizzare questo modello per raggiungere i tuoi obiettivi di modernizzazione delle applicazioni o di analisi dei dati. Ad esempio, puoi utilizzare Connect per migrare i file dell'applicazione VSAM sul cloud AWS con bassa latenza o migrare i dati VSAM verso un data warehouse o un data lake AWS per analisi in grado di tollerare latenze di sincronizzazione superiori a quelle richieste per la modernizzazione delle applicazioni.

Prerequisiti e limitazioni

Prerequisiti

- IBM V2R1 o versione successiva z/OS
- CICS Transaction Server per z/OS (CICS TS) V5.1 o versione successiva (acquisizione dati CICS/VSAM)
- IBM MQ 8.0 o versione successiva
- Conformità ai requisiti di sicurezza z/OS (ad esempio, autorizzazione APF per le librerie di caricamento) SQData
- I log di ripristino VSAM sono attivati
- (Opzionale) CICS VSAM Recovery Version (CICS VR) per acquisire automaticamente i registri CDC
- Un account AWS attivo
- Un Amazon Virtual Private Cloud (VPC) con una sottorete raggiungibile dalla tua piattaforma legacy
- Una licenza VSAM Connect di Precisely

Limitazioni

- Connect non supporta la creazione automatica di tabelle di destinazione basate su schemi o quaderni VSAM di origine. È necessario definire la struttura della tabella di destinazione per la prima volta.
- Per destinazioni non in streaming come Amazon RDS, è necessario specificare la mappatura tra origine di conversione e destinazione nello script di configurazione di Apply Engine.
- Le funzioni di registrazione, monitoraggio e avviso sono implementate APIs e richiedono componenti esterni (come Amazon CloudWatch) per essere completamente operativi.

Versioni del prodotto

- SQData 40134 per z/OS
- SQData 4.0.43 per Amazon Linux Amazon Machine Image (AMI) su Amazon Elastic Compute Cloud (Amazon) EC2

Architecture

Stack tecnologico di origine

- Job Control Language (JCL)
- Shell Unix z/OS e Interactive System Productivity Facility (ISPF)
- Utilità VSAM (IDCAMS)

Stack tecnologico Target

- Amazon EC2
- MSK Amazon
- Amazon RDS
- Amazon VPC

Architettura Target

Migrazione di file VSAM su Amazon RDS

Il diagramma seguente mostra come migrare i file VSAM su un database relazionale, come Amazon RDS, in tempo reale o quasi reale utilizzando il CDC nell'ambiente di origine (mainframe locale) e l'Apply Engine agent/publisher nell'ambiente di destinazione (AWS Cloud).

Il diagramma mostra il seguente flusso di lavoro in batch:

1. Connect acquisisce le modifiche a un file confrontando i file VSAM dai file di backup per identificare le modifiche e quindi invia le modifiche al logstream.
2. L'editore utilizza i dati dal logstream di sistema.
3. L'editore comunica le modifiche ai dati acquisiti a un motore di destinazione tramite TCP/IP. Il Controller Daemon autentica la comunicazione tra l'ambiente di origine e quello di destinazione.
4. Il motore di applicazione nell'ambiente di destinazione riceve le modifiche dall'agente Publisher e le applica a un database relazionale o non relazionale.

Il diagramma mostra il seguente flusso di lavoro online:

1. Connect acquisisce le modifiche nel file online utilizzando una replica del registro e quindi trasmette le modifiche acquisite in un flusso di registro.
2. L'editore utilizza i dati dal logstream di sistema.
3. L'editore comunica le modifiche ai dati acquisiti al motore di destinazione tramite TCP/IP. Il Controller Daemon autentica la comunicazione tra l'ambiente di origine e quello di destinazione.
4. Il motore di applicazione nell'ambiente di destinazione riceve le modifiche dall'agente Publisher e quindi le applica a un database relazionale o non relazionale.

Migrazione di file VSAM su Amazon MSK

Il diagramma seguente mostra come eseguire lo streaming di strutture di dati VSAM da un mainframe ad Amazon MSK in modalità ad alte prestazioni e generare automaticamente conversioni di schemi JSON o AVRO che si integrano con Amazon MSK.

Il diagramma mostra il seguente flusso di lavoro in batch:

1. Connect acquisisce le modifiche a un file utilizzando CICS VR o confrontando i file VSAM dai file di backup per identificare le modifiche. Le modifiche acquisite vengono inviate al logstream.

2. L'editore utilizza i dati dal logstream di sistema.
3. L'editore comunica le modifiche ai dati acquisiti al motore di destinazione tramite TCP/IP. Il Controller Daemon autentica la comunicazione tra l'ambiente di origine e quello di destinazione.
4. Il Replicator Engine che opera in modalità di elaborazione parallela divide i dati in un'unità di cache di lavoro.
5. I thread di lavoro acquisiscono i dati dalla cache.
6. I dati vengono pubblicati sugli argomenti di Amazon MSK dai thread di lavoro.
7. Gli utenti applicano le modifiche da Amazon MSK a destinazioni come Amazon DynamoDB, Amazon Simple Storage Service (Amazon S3) OpenSearch o Amazon Service utilizzando i connettori.

Il diagramma mostra il seguente flusso di lavoro online:

1. Le modifiche nel file online vengono acquisite utilizzando una replica del registro. Le modifiche acquisite vengono trasmesse al logstream.
2. L'editore utilizza i dati dal logstream di sistema.
3. L'editore comunica le modifiche ai dati acquisiti al motore di destinazione tramite TCP/IP. Il Controller Daemon autentica la comunicazione tra l'ambiente di origine e quello di destinazione.
4. Il Replicator Engine che opera in modalità di elaborazione parallela divide i dati in un'unità di cache di lavoro.
5. I thread di lavoro acquisiscono i dati dalla cache.
6. I dati vengono pubblicati sugli argomenti di Amazon MSK dai thread di lavoro.
7. Gli utenti applicano le modifiche da Amazon MSK a destinazioni come DynamoDB, Amazon S3 o Service utilizzando i connettori OpenSearch .

Tools (Strumenti)

- Amazon Managed Streaming for Apache Kafka (Amazon MSK) è un servizio completamente gestito che ti aiuta a creare ed eseguire applicazioni che utilizzano Apache Kafka per elaborare dati di streaming.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS.

Epiche

Preparare l'ambiente di origine (mainframe)

Operazione	Description	Competenze richieste
Installare Connect CDC 4.1.	1. Contatta il team di Precisly Support per ottenere una licenza e i pacchetti di installazione. 2. Usa l'esempio JCLs per installare Connect CDC 4.1. Per istruzioni, vedete Install Connect CDC (SQData) usando JCL nella documentazione di Precisly. 3. Esegui il `SETPROG APF` comando per autorizza re le librerie di caricamen to Connect sqdata.v4 nnn.loadLib.	Sviluppatore/amministratore IBM Mainframe
Configura la directory ZfS.	Per configurare una directory ZfS, seguite le istruzioni contenute nelle directory delle variabili ZFs nella documenta zione di Precisly. **Note** Le configurazioni dei controller daemon e degli Capture/P ublisher agenti sono memorizzate nel file system z/OS UNIX Systems Services	Sviluppatore/amministratore IBM Mainframe

Operazione	Description	Competenze richieste
	(denominato ZFs). Gli agenti Controlle r Daemon, Capture, Storage e Publisher richiedono una struttura di directory ZfS predefinita per archiviare un numero limitato di file.	
TCP/IP Configura le porte.	Per configurare le TCP/IP porte, segui le istruzioni fornite dalle porte TCP/IP nella documentazione di Precisly. **Note** Il Controller Daemon richiede porte sui sistemi di origine TCP/IP . Alle porte fanno riferimento i motori dei sistemi di destinazi one (dove vengono elaborati i dati di modifica acquisiti).	Sviluppatore/amministratore IBM Mainframe

Operazione	Description	Competenze richieste
Crea un z/OS logstream.	Per creare un logstream z/OS, segui le istruzioni di Create z/OS system LogStreams nella documentazione di Precisly. **Note** Connect utilizza il logstream per acquisire e trasmette re dati tra l'ambiente di origine e l'ambiente di destinazione durante la migrazione. Per un esempio JCL che crea un z/OS LogStream, consulta Create z/OS System LogStreams nella documenta zione di Precisly.	Sviluppatore IBM Mainframe
Identifica e autorizza gli utenti ZfS e IDs le attività avviate.	Utilizzate RACF per concedere l'accesso al file system OMVS ZfS. Per un esempio JCL, vedete Identific are e autorizzare l'utente e l'attività avviata da ZfS nella documentazione di Precisly. IDs	Sviluppatore/amministratore IBM Mainframe

Operazione	Description	Competenze richieste
Genera chiavi z/OS pubbliche /private e il file di chiave autorizzato.	Esegui JCL per generare la key pair. Per un esempio, vedi Esempio di coppia di chiavi nella sezione Informazioni aggiuntive di questo modello. Per istruzioni, consulta Generare chiavi z/OS pubbliche e private e file di chiavi autorizzate nella documentazione di Precisly.	Sviluppatore/amministratore di IBM Mainframe

Attiva CICS VSAM Log Replicate e collegalo al logstream.	Esegui il seguente script JCL:
	
	`//STEP1 EXEC PGM=IDCAMS`
	`//SYSPRINT DD SYSOUT=*`
	`//SYSIN DD *`
	`ALTER SQDATA.CICS.FILEA -`
	`LOGSTREAMID(SQDATA.VSAMCDC.LOG1) -`
	`LOGREPLICATE`
		Sviluppatore/amministratore IBM Mainframe

Operazione	Description	Competenze richieste
Attiva il registro di ripristino dei file VSAM tramite un FCT.	Modificate la File Control Table (FCT) in modo che rifletta le seguenti modifiche ai parametri: `Configure FCT Parms CEDA ALT FILE(name) GROUP(groupname) DSNAME(data set name) RECOVERY(NONE|BACKOUTONLY|ALL) FWDRECOVLOG(NO|1-99) BACKUPTYPE(STATIC|DYNAMIC) RECOVERY PARAMETERS RECOVery : None | Backoutonly | All Fwdrecovlog : No | 1-99 BAckuptype : Static | Dynamic`	Sviluppatore/amministratore IBM Mainframe
Configura CDCz Log per l'agente Publisher.	1. Creare il file CAB di CDCz Log Publisher. 2. Crittografa i dati pubblicati. 3. Preparare il JCL di CDCz Log Publisher Runtime.	Sviluppatore/amministratore IBM Mainframe

Operazione	Description	Competenze richieste
Attiva il Controller Daemon.	1. Aprite il pannello ISPF ed eseguite il seguente comando per aprire il menu Precisamente: `EXEC 'SQDATA.V4nnnnn.ISPFLIB(SQDC$STA)' 'SQDATA.V4nnnnn'` 2. Per configurare il Controller Daemon, scegliete l'opzione 2 dal menu.	Sviluppatore/amministratore IBM Mainframe
Attiva l'editore.	1. Aprite il pannello ISPF ed eseguite il seguente comando per aprire il menu Precisamente: `EXEC 'SQDATA.V4nnnnn.ISPFLIB(SQDC$STA)' 'SQDATA.V4nnnnn'` 2. Per configurare l'editore , scegliete l'opzione 3 dal menu e io per l'inserimento.	Sviluppatore/amministratore IBM Mainframe

Operazione	Description	Competenze richieste
Attiva il logstream.	1. Aprite il pannello ISPF ed eseguite il seguente comando per aprire il menu Precisamente: `EXEC 'SQDATA.V4nnnnn.ISPFLIB(SQDC$STA)' 'SQDATA.V4nnnnn'` 2. Per configurare il logstream, scegliete l'opzione 4 dal menu e io per l'inserimento. Quindi, inserisci il nome del logstream creato nei passaggi precedenti.	Sviluppatore/amministratore IBM Mainframe

Preparare l'ambiente di destinazione (AWS)

Operazione	Description	Competenze richieste
Installa Precisly su un' EC2 istanza.	Per installare Connect from Precisly sull'AMI Amazon Linux per Amazon EC2, segui le istruzioni di Install Connect CDC (SQData) su UNIX nella documentazione di Precisly.	Informazioni generali su AWS
TCP/IP Porte aperte.	Per modificare il gruppo di sicurezza in modo da includere le porte Controller Daemon per l'accesso in entrata e in uscita, segui le istruzioni di TCP/IP nella documentazione di Precisly.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Crea directory di file.	Per creare directory di file, segui le istruzioni di Prepare target apply environment nella documentazione di Precisly.	Informazioni generali su AWS

Crea il file di configurazione di Apply Engine.	Create il file di configurazione di Apply Engine nella directory di lavoro di Apply Engine. Il seguente file di configurazione di esempio mostra Apache Kafka come destinazione:
	`builtin.features=SASL_SCRAM`
	`security.protocol=SASL_SSL`
	`sasl.mechanism=SCRAM-SHA-512`
	`sasl.username=`
	`sasl.password=`
	`metadata.broker.list=`
	Note
	Per ulteriori informazioni, consulta Sicurezza nella documentazione di Apache Kafka.
		Informazioni generali su AWS

Operazione	Description	Competenze richieste
Crea script per l'elaborazione di Apply Engine.	Create gli script per Apply Engine per elaborare i dati di origine e replicare i dati di origine sulla destinazione. Per ulteriori informazioni, vedete Creare uno script del motore di applicazione nella documentazione di Precisly.	Informazioni generali su AWS
Esegui gli script.	Utilizzate i `SQDENG` comandi `SQDPARSE` and per eseguire lo script. Per ulteriori informazioni, consulta Analizzare uno script per zOS nella documentazione di Precisly.	Informazioni generali su AWS

Convalida l'ambiente

Operazione	Description	Competenze richieste
Convalida l'elenco dei file VSAM e delle tabelle di destinazione per l'elaborazione CDC.	1. Convalida i file VSAM, inclusi i log di replica, i log di ripristino, i parametri FCT e il logstream. 2. Convalida le tabelle del database di destinazione, specificando se le tabelle vengono create in base alla definizione dello schema richiesta, all'accesso alla tabella e ad altri criteri.	Informazioni generali su AWS, mainframe

Operazione	Description	Competenze richieste
Verificare che il SQData prodotto Connect CDC sia collegato.	Esegui un processo di test e verifica che il codice restituit o da questo lavoro sia 0 (operazione riuscita). **Note** I messaggi di stato di Connect CDC SQData Apply Engine dovrebbero mostrare messaggi di connessio ne attivi.	Informazioni generali su AWS, mainframe

Esecuzione e convalida dei casi di test (Batch)

Operazione	Description	Competenze richieste
Esegui il processo batch nel mainframe.	Esegui il processo di applicazi one batch utilizzando un JCL modificato. Includi i passaggi nel JCL modificat o che eseguono le seguenti operazioni: 1. Effettua un backup dei file di dati. 2. Confronta il file di backup con i file di dati modificati, genera il file delta, quindi annota il conteggio dei record delta dei messaggi. 3. Invia il file delta al z/OS logstream.	Informazioni generali su AWS, mainframe

Operazione	Description	Competenze richieste
	4. Esegui il JCL. Per un esempio JCL, consulta Prepare file compare capture JCL nella documentazione di Precisly.	
Controlla il logstream.	Controllate il logstream per confermare che è possibile visualizzare i dati di modifica per il processo batch mainframe completato.	Informazioni generali su AWS, mainframe
Convalida i conteggi per le modifiche delta di origine e la tabella di destinazione.	Per confermare il conteggio dei record, procedi come segue: 1. Raccogli il conteggio delta di origine dai messaggi JCL in batch. 2. Monitora il motore di applicazione per il conteggio a livello di record del numero di record inseriti, aggiornati o eliminati nel file VSAM. 3. Interroga la tabella di destinazione per il conteggio dei record. 4. Confronta e calcola tutti i diversi conteggi dei record.	Informazioni generali su AWS, mainframe

Esegui e convalida i casi di test (online)

Operazione	Description	Competenze richieste
Esegui la transazione online in una regione CICS.	1. Esegui la transazione online per convalidare il test case. 2. Convalida il codice di esecuzione della transazio ne (RC=0 — Success).	Sviluppatore IBM Mainframe
Controlla il logstream.	Verifica che il logstream sia popolato con modifiche specifiche al livello di record.	Sviluppatore di mainframe AWS
Convalida il conteggio nel database di destinazione.	Monitora Apply Engine per conteggi a livello record.	Precisamente, Linux
Convalida il conteggio dei record e i record di dati nel database di destinazione.	Interroga il database di destinazione per convalidare il numero di record e i record di dati.	Informazioni generali su AWS

Risorse correlate

- VSAM z/OS (documentazione precisa)
- Applica il motore (documentazione precisa)
- Motore Replicator (documentazione precisa)
- Il flusso di log (documentazione IBM)

Informazioni aggiuntive

Esempio di file di configurazione

Questo è un esempio di file di configurazione per un logstream in cui l'ambiente di origine è un mainframe e l'ambiente di destinazione è Amazon MSK:

```
 -- JOBNAME -- PASS THE SUBSCRIBER NAME
 -- REPORT  progress report will be produced after "n" (number) of Source records
processed.

 JOBNAME VSMTOKFK;
 --REPORT EVERY 100;
 -- Change Op has been ‘I’ for insert, ‘D’ for delete , and ‘R’ for Replace. For RDS
it is 'U' for update
 -- Character Encoding on z/OS is Code Page 1047, on Linux and UNIX it is Code Page
819 and on Windows, Code Page 1252
 OPTIONS
 CDCOP('I', 'U', 'D'),
 PSEUDO NULL = NO,
 USE AVRO COMPATIBLE NAMES,
 APPLICATION ENCODING SCHEME = 1208;

 --       SOURCE DESCRIPTIONS

 BEGIN GROUP VSAM_SRC;
 DESCRIPTION COBOL ../copybk/ACCOUNT AS account_file;
 END GROUP;

 --       TARGET DESCRIPTIONS

 BEGIN GROUP VSAM_TGT;
 DESCRIPTION COBOL ../copybk/ACCOUNT AS account_file;
 END GROUP;

 --       SOURCE DATASTORE (IP & Publisher name)

 DATASTORE cdc://10.81.148.4:2626/vsmcdct/VSMTOKFK
 OF VSAMCDC
 AS CDCIN
 DESCRIBED BY GROUP VSAM_SRC ACCEPT ALL;

 --       TARGET DATASTORE(s) - Kafka and topic name

 DATASTORE 'kafka:///MSKTutorialTopic/key'
 OF JSON
 AS CDCOUT
 DESCRIBED BY GROUP VSAM_TGT FOR INSERT;

 --       MAIN SECTION
```

```
  PROCESS INTO
  CDCOUT
  SELECT
  {
  SETURL(CDCOUT, 'kafka:///MSKTutorialTopic/key')
  REMAP(CDCIN, account_file, GET_RAW_RECORD(CDCIN, AFTER), GET_RAW_RECORD(CDCIN,
 BEFORE))
  REPLICATE(CDCOUT, account_file)
  }
  FROM CDCIN;
```

Esempio di coppia di chiavi

Questo è un esempio di come eseguire JCL per generare la key pair:

```
//SQDUTIL EXEC PGM=SQDUTIL //SQDPUBL DD DSN=&USER..NACL.PUBLIC, //
 DCB=(RECFM=FB,LRECL=80,BLKSIZE=21200), // DISP=(,CATLG,DELETE),UNIT=SYSDA, //
 SPACE=(TRK,(1,1)) //SQDPKEY DD DSN=&USER..NACL.PRIVATE, //
 DCB=(RECFM=FB,LRECL=80,BLKSIZE=21200), // DISP=(,CATLG,DELETE),UNIT=SYSDA, //
 SPACE=(TRK,(1,1)) //SQDPARMS DD  keygen //SYSPRINT DD SYSOUT= //SYSOUT DD SYSOUT=* //
SQDLOG DD SYSOUT=* //*SQDLOG8 DD DUMMY
```

Modernizza l'applicazione CardDemo mainframe utilizzando AWS Transform

Santosh Kumar Singh e Cheryl du Preez, Amazon Web Services

Riepilogo

AWS Transformè progettato per accelerare la modernizzazione delle applicazioni mainframe. Utilizza l'intelligenza artificiale generativa per semplificare il processo di modernizzazione del mainframe. Automatizza attività complesse, come l'analisi del codice legacy, la documentazione del mainframe, l'estrazione delle regole aziendali, la scomposizione delle applicazioni monolitiche in un dominio aziendale e il refactoring del codice. Accelera i progetti di modernizzazione automatizzando attività complesse, come l'analisi delle applicazioni e la pianificazione delle sequenze di migrazione. Durante la scomposizione di applicazioni monolitiche, AWS Transform sequenzia in modo intelligente la trasformazione delle applicazioni mainframe, il che aiuta a trasformare le funzioni aziendali in parallelo. AWS Transform può accelerare il processo decisionale e migliorare l'agilità operativa e l'efficienza della migrazione.

Questo modello offre step-by-step istruzioni per aiutarvi a testare le capacità di modernizzazione del mainframe AWS Transform utilizzando CardDemo, che è un esempio di applicazione mainframe open source.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- AWS IAM Identity Center, abilitato
- Autorizzazioni che consentono agli amministratori di abilitare AWS Transform
- Autorizzazioni che consentono agli amministratori di accettare le richieste di connessione Amazon Simple Storage Service (Amazon S3) per l'applicazione Web AWS Transform

Limitazioni

- AWS Transform è disponibile solo in alcuni. Regioni AWS Per un elenco completo delle regioni supportate, consulta Regioni supportate per AWS Transform.
- AWS Transform supporta l'analisi del codice, la generazione di documenti, l'estrazione, la scomposizione e il refactoring delle regole aziendali da Common Business-Oriented Language

(COBOL) a Java. Per ulteriori informazioni, vedere Funzionalità e caratteristiche chiave e Tipi di file supportati per la trasformazione delle applicazioni mainframe.

- È prevista una quota di servizio per le funzionalità di trasformazione del mainframe in. AWS Transform Per ulteriori informazioni, vedere Quotas for. AWS Transform
- Per collaborare su uno spazio di lavoro condiviso, tutti gli utenti devono essere utenti registrati della stessa istanza associata all'istanza dell' AWS Transform applicazione Web. AWS IAM Identity Center
- Il bucket Amazon S3 e AWS Transform devono trovarsi nella stessa Account AWS regione.

Architecture

Il diagramma seguente mostra l'architettura configurata secondo questo modello.

Il diagramma mostra il flusso di lavoro seguente:

1. AWS Transform utilizza un connettore per accedere all'applicazione CardDemo mainframe, archiviata in un bucket Amazon S3.
2. AWS Transform utilizza AWS IAM Identity Center per gestire l'accesso e l'autenticazione degli utenti. Il sistema implementa più livelli di controlli di sicurezza per l'autenticazione, l'autorizzazione, la crittografia e la gestione degli accessi per proteggere codice e artefatti durante l'elaborazione. Gli utenti interagiscono con l' AWS Transform agente tramite un'interfaccia di chat. Puoi fornire istruzioni all'agente AI per attività specifiche in inglese. Per ulteriori informazioni, consulta Human in the loop (HITL) nella AWS Transform documentazione.
3. L'agente AI interpreta le istruzioni dell'utente, crea un piano di lavoro, divide il lavoro in attività eseguibili e lo esegue in modo autonomo. Gli utenti possono rivedere e approvare la trasformazione. Le attività di trasformazione includono quanto segue:
   - Analisi del codice: AWS Transform analizza il codice contenuto in ogni file alla ricerca di dettagli quali il nome del file, il tipo di file, le righe di codice e i relativi percorsi. L'agente analizza il codice sorgente, esegue classificazioni, crea mappature delle dipendenze e identifica eventuali artefatti mancanti. Identifica inoltre i componenti duplicati.
   - Generazione di documenti: AWS Transform genera documentazione per l'applicazione mainframe. Analizzando il codice, può creare automaticamente una documentazione dettagliata dei programmi applicativi, comprese le descrizioni della logica aziendale, dei flussi, delle integrazioni e delle dipendenze presenti nei sistemi legacy.

- Estrazione della logica aziendale: AWS Transform analizza i programmi COBOL per documentarne la logica aziendale principale, per aiutarvi a comprendere la logica aziendale fondamentale.
- Scomposizione del codice: AWS Transform scompone il codice in domini che tengono conto delle dipendenze tra programmi e componenti. Il raggruppamento di file e programmi correlati all'interno dello stesso dominio migliora l'organizzazione e aiuta a preservare la struttura logica dell'applicazione quando la scompone in componenti più piccoli.
- Pianificazione dell'ondata di migrazione: in base ai domini creati durante la fase di decomposizione, AWS Transform genera un piano di ondata di migrazione con l'ordine di modernizzazione consigliato.
- Rifattorizzazione del codice: AWS Transform rifattorizza il codice in tutti i file di dominio o in alcuni file di dominio in codice Java. L'obiettivo di questo passaggio è preservare la logica aziendale fondamentale dell'applicazione rifattorizzandola in un'applicazione Java modernizzata e ottimizzata per il cloud.

4. AWS Transform archivia il codice rifattorizzato, i documenti generati, gli artefatti associati e le librerie di runtime nel tuo bucket Amazon S3. Puoi fare quanto segue:
   - Accedi alla cartella runtime nel tuo bucket Amazon S3.
   - Crea e distribuisci l'applicazione seguendo le istruzioni Compila e distribuisci la tua applicazione modernizzata dopo il refactoring riportato nella documentazione. AWS Transform
   - Tramite l'interfaccia di chat, richiedi e scarica un esempio o un modello Hashicorp AWS CloudFormation Terraform AWS Cloud Development Kit (AWS CDK). Questi modelli possono aiutarti a distribuire AWS le risorse necessarie per supportare l'applicazione refactorizzata.
   - Utilizzate Reforge per migliorare la qualità del codice rifattorizzato utilizzando modelli di linguaggio di grandi dimensioni (). LLMs Il motore di refactoring preserva l'equivalenza funzionale di COBOL trasformandolo in codice Java. Reforge è un passaggio opzionale disponibile dopo la trasformazione. Questo passaggio serve LLMs a ristrutturare il codice in modo che assomigli da vicino a Java nativo, il che può migliorare la leggibilità e la manutenibilità. Reforge aggiunge anche commenti leggibili dall'uomo per aiutarti a comprendere il codice e implementa modelli di codifica e best practice moderni.

Tools (Strumenti)

Servizi AWS

- AWS Transformutilizza l'intelligenza artificiale agentica per aiutarti ad accelerare la modernizzazione dei carichi di lavoro legacy, come .NET, mainframe e carichi di lavoro. VMware
- AWS IAM Identity Centerti aiuta a gestire centralmente l'accesso Single Sign-On (SSO) alle tue applicazioni e a quelle cloud. Account AWS
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Archivio di codice

È possibile utilizzare l'applicazione AWS CardDemomainframe open source come applicazione di esempio per iniziare con la modernizzazione del mainframe.

Best practice

- Inizia in piccolo: inizia con codice piccolo e meno complesso (15.000—20.000 righe di codice) per comprendere come analizza e trasforma le applicazioni mainframe. AWS Transform
- Combina con l'esperienza umana: utilizza AWS Transform come acceleratore mentre applichi l'esperienza umana per risultati ottimali.
- Revisione e test accurati: rivedi sempre attentamente il codice trasformato ed esegui test completi per convalidare l'equivalenza funzionale dopo la trasformazione.
- Fornisci feedback: per fornire feedback e suggerimenti per il miglioramento, utilizza il pulsante Invia feedback in Console di gestione AWS o crea un caso con. Supporto AWS Per ulteriori informazioni, consulta Creazione di un caso di supporto. Il tuo contributo è prezioso per il miglioramento del servizio e lo sviluppo futuro.

Epiche

Preparare l'applicazione mainframe

Operazione	Description	Competenze richieste
Creare un bucket.	Crea un bucket Amazon S3 nella stessa regione in cui AWS Transform è Account AWS abilitato. Questo bucket viene utilizzato per	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	archiviare il codice dell'appl icazione mainframe e lo AWS Transform utilizza per archiviar e i documenti generati, il codice refactorizzato e altri file associati alla trasforma zione. Per istruzioni, consulta Creazione di un bucket nella documentazione di Amazon S3.	

Operazione	Description	Competenze richieste
Preparare l'applicazione mainframe di esempio.	1. Immettete il seguente comando per clonare il CardDemorepository sulla workstation locale: `git clone https://github.com/aws-samples/aws-mainframe-modernization-carddemo.git` 2. Crea una nuova cartella denominata. `carddemo` 3. Copiate la `app` cartella, che contiene il codice sorgente del mainframe, dal repository clonato nella cartella. `carddemo` 4. Comprimi la `cardemo` cartella in un file ZIP. 5. Carica il file ZIP nel bucket Amazon S3 che hai creato. Per istruzioni, consulta Caricamento di oggetti nella documentazione di Amazon S3.	Sviluppatore di app, ingegnere DevOps

Configura IAM Identity Center e AWS Transform

Operazione	Description	Competenze richieste
Aggiungi utenti a IAM Identity Center.	Aggiungi i tuoi potenziali utenti a IAM Identity Center. Segui le istruzioni in Aggiungere utenti	Amministratore AWS

Operazione	Description	Competenze richieste
	in IAM Identity Center nella AWS Transform documenta zione.	
Abilita AWS Transform e aggiungi utenti.	1. Abilita AWS Transform. Segui le istruzioni riportate in Abilita AWS Transform. 2. Aggiungi utenti a AWS Transform. Segui le istruzioni riportate in Aggiungere utenti a AWS Transform.	Amministratore AWS
Configura l'accesso degli utenti all'applicazione AWS Transform web.	Ogni utente deve accettare l'invito ad accedere all'appli cazione AWS Transform web. Segui le istruzioni riportate in Accettazione dell'invito nella AWS Transform documenta zione.	Sviluppatore dell'app, proprietario dell'app
Accedi all'applicazione AWS Transform web.	Segui le istruzioni riportate in Accesso a AWS Transform.	Sviluppatore dell'app, proprietario dell'app
Configura uno spazio di lavoro.	Configura uno spazio di lavoro in cui gli utenti possano collaborare nell' AWS Transform applicazione web. Segui le istruzioni riportate in Avvia il tuo progetto nella AWS Transform documenta zione.	Amministratore AWS

Trasforma l'applicazione mainframe

Operazione	Description	Competenze richieste
Crea un lavoro di trasforma zione.	Crea un processo di trasforma zione per modernizzare l'applicazione CardDemo mainframe. Per istruzioni, consulta Creare e avviare un lavoro nella AWS Transform documentazione. Quando ti viene chiesto di impostare gli obiettivi nell'interfaccia di AWS Transform chat, scegli Esegui la modernizzazione del mainframe (da IBM z/ OS a AWS), quindi scegli Analizza codice, Genera documentazione tecnica, Logica aziendale, Scomponi codice, Pianifica la sequenza di migrazione e Trasforma il codice in Java.	Sviluppatore di app, proprieta rio dell'app
Configura il connettore.	Stabilisci un connettore per il bucket Amazon S3 che contiene l' CardDemo applicazione mainframe. Questo connettore consente di accedere AWS Transform alle risorse nel bucket ed eseguire funzioni di trasforma zione consecutive. Per istruzioni, consulta Configura re un connettore nella AWS Transform documentazione.	Amministratore AWS

Operazione	Description	Competenze richieste
Esegui l'analisi del codice.	1. Nella pagina Specifica re la posizione dell'asse t, inserisci il percorso del bucket Amazon S3 per il file `carddemo` ZIP che hai caricato. 2. Scegli Approva e invia a. AWS Transform Inizia ad analizzare il codice. 3. Monitora lo stato nella scheda Worklog. 4. Una volta completata l'analisi, in Analizza il codice nel riquadro di navigazione a sinistra, scegli Visualizza i risultati dell'analisi del codice. 5. (Facoltativo) Scegliete Scarica per scaricare l'elenco completo delle risorse, il codice sorgente mancante e il file di dipendenza. Per ulteriori informazioni, consultate Analisi del codice nella AWS Transform documentazione.	Sviluppatore di app, proprieta rio dell'app

Operazione	Description	Competenze richieste
Genera documentazione tecnica.	1. Nel riquadro di navigazione a sinistra, in Genera documentazione tecnica, scegli Seleziona file e configura le impostazioni. 2. Espandi COBOL o JCL, quindi seleziona uno o più file. 3. Scegliete il livello di dettaglio della documentazione: • Riepilogo: fornisce una panoramica di alto livello di ogni file compreso nell'ambito. Inoltre, fornisce un riepilogo di ogni file in una riga. • Specifiche funzionali dettagliate: fornisce dettagli completi per ogni file nell'ambito della trasformazione delle applicazioni mainframe. Alcuni dettagli includono logica e flusso, regole aziendali identificate, flusso di dati, dipendenze, elaborazione di input e output e vari dettagli sulle transazioni. 4. Scegli Invia a. AWS Transform	Sviluppatore di app, proprietario dell'app

Operazione	Description	Competenze richieste
	5. Monitora l'avanzamento nella scheda Worklog. > **Note** > Il tempo di generazione dei documenti varia da minuti a ore, a seconda del numero di file e righe di codice. 6. Al termine, scegli Rivedi i risultati della documenta zione. 7. Accedi al documento generato tramite l'applica zione web o scaricalo. La documentazione generata viene inoltre archiviata nel bucket Amazon S3. 8. Esplora la documenta zione generata ponendo domande specifiche nell'interfaccia di chat. Ad esempio, puoi dire «Dimmi le caratteristiche di CBACT01 C.» Per ulteriori informazioni, consulta Generare documenta zione tecnica nella AWS Transform documentazione.	

Operazione	Description	Competenze richieste
Estrai la logica aziendale.	1. Nel riquadro di navigazione a sinistra, espandi Estrai la logica aziendale, quindi scegli Seleziona i file per l'estrazione della logica aziendale. 2. Espandi COBOL o JCL, quindi seleziona uno o più file. 3. Scegliete Invia a. AWS Transform 4. Monitora l'avanzamento nella scheda Worklog. **Note** Il tempo di estrazione della logica aziendale varia da minuti a ore, a seconda del numero di file e righe di codice. 5. Quando l'estrazione è completa, scegli Rivedi i risultati dell'estrazione della logica aziendale. 6. Selezionate un file COBOL o JCL, quindi scegliete Visualizza. 7. Modificate l'ambito nel diagramma di flusso e nelle regole funzionali per	Sviluppatore di app, proprietario dell'app

Operazione	Description	Competenze richieste
	esaminare i diversi livelli dei diagrammi di flusso, incluso il codice di supporto. 8. (Facoltativo) Esamina la logica di business generata, archiviata in formato JSON nel tuo bucket Amazon S3. Per ulteriori informazioni, consulta Extract business logic nella documentazione. AWS Transform	

Operazione	Description	Competenze richieste
Decomponi il codice.	1. Nel riquadro di navigazione a sinistra, espandi Decomponi codice, quindi scegli Scomponi in domini. 2. Nell'elenco Azioni, scegli Crea dominio. 3. Inserisci un nome per il nuovo dominio, ad esempio`Bill payment`. 4. (Facoltativo) Fornisci una descrizione. 5. Nella barra di ricerca Trova file`CB00`, cerca e seleziona il file. 6. Scegli Contrassegna come seme. 7. Verifica che il flag del seme cambi da No a Sì. 8. Scegli Create (Crea). 9. Scegli Save (Salva). 10. Scegliete Decompose. 11. Controlla l'output di scomposizione scegliendo il nome di dominio. 12. Quando la scomposizione è completa, scegli Invia a. AWS Transform Per ulteriori informazioni sulla decomposizione e sui semi, vedete Decomposizione	Sviluppatore di app, proprietario dell'app

Operazione	Description	Competenze richieste
	nella documentazione. AWS Transform	
Pianifica le ondate migratorie.	Pianifica le ondate di migrazione per l' CardDemo applicazione. Segui le istruzioni riportate nella pianificazione delle ondate di migrazione nella AWS Transform documentazione per rivedere e modificare il piano delle ondate.	Sviluppatore dell'app, proprietario dell'app
Rifattorizza il codice.	1. Effettua il refactoring dell'applicazione CardDemo mainframe in codice Java in tutti i file di dominio o in alcuni file di dominio. Segui le istruzioni contenute nel codice Refactor nella documentazione. AWS Transform 2. Una volta completato il processo di refactoring, create e distribuite l'applicazione Java modernizzata. Segui le istruzioni riportate in Crea e distribuisci la tua applicazione modernizzata dopo il refactoring nella documentazione. AWS Transform	Sviluppatore di app, proprietario dell'app

Operazione	Description	Competenze richieste
(Facoltativo) Usa Reforge per migliorare il codice Java.	1. Crea un nuovo lavoro, quindi inserisci un obiettivo di. `Reforge the code` 2. Inserisci il nome di un progetto che è stato modernizzato. AWS Transform Questo progetto dovrebbe essere nel tuo bucket Amazon S3. 3. Inserisci l'elenco delle classi Java che specifica le classi di servizio da riforgiare. 4. Esamina l'output generato nel bucket Amazon S3. Per ulteriori informazioni, consulta Reforge nella documentazione. AWS Transform	Sviluppatore dell'app, proprietario dell'app

Operazione	Description	Competenze richieste
Semplifica l'implementazione.	AWS Transform può fornire modelli Infrastructure as Code (IaC) per CloudFormation AWS CDK Terraform. Questi modelli consentono di distribuire i componenti principali, tra cui risorse di elaborazione, database, archiviazione e sicurezza. 1. Nell'interfaccia di AWS Transform chat, richiedi i modelli inserendo `Share the mainframe refactor <service> templates` , `<service>` dov'è uno dei servizi IaC supportati. 2. Implementa il modello utilizzando il tuo servizio preferito. Ad esempio, consulta le seguenti risorse: • Crea uno stack dalla CloudFormation console (CloudFormation documentazione) • Distribuisci AWS CDK applicazioni (documentazione)AWS CDK • Crea un'infrastruttura (documentazione Terraform)	Sviluppatore di app, proprietario dell'app

Operazione	Description	Competenze richieste
	Per ulteriori informazioni, consulta le funzionalità di distribuzione nella AWS Transform documentazione.	

Risoluzione dei problemi

Problema	Soluzione
Non è possibile visualizzare il codice sorgente o il documento generato nell'applicazione AWS Transform Web.	Aggiungi una policy all'autorizzazione CORS per il bucket Amazon S3 da AWS Transform consentire come origine. Per ulteriori informazioni, consulta le autorizzazioni CORS del bucket S3 nella documentazione. AWS Transform

Risorse correlate

AWS documentazione

- Trasformazione delle applicazioni mainframe (AWS Transform documentazione)

Altre risorse AWS

- Accelera il tuo percorso di modernizzazione del mainframe utilizzando agenti AI con AWS Transform (AWS post sul blog)
- AWS Transform FAQs
- AWS IAM Identity Center FAQs

Video e tutorial

- Introduzione ad Amazon Q Developer: Transform (AWS Skill Builder)
- AWS re:Invent 2024 - Modernizza più rapidamente le applicazioni mainframe con Amazon Q Developer () YouTube

- AWS re:Invent 2024 - Automatizzazione della migrazione e della modernizzazione per accelerare la trasformazione () YouTube
- AWS re:Invent 2024 - Toyota promuove l'innovazione e migliora l'efficienza operativa con gen AI () YouTube

> **Note**
>
> AWS Transform era precedentemente noto come Amazon Q Developer transform for mainframe.

Modernizza e implementa le applicazioni mainframe utilizzando e Terraform AWS Transform

Mason Cahill, Polaris Jhandi, Prachi Khanna, Sivasubramanian Ramani e Santosh Kumar Singh, Amazon Web Services

Riepilogo

AWS Transformpuò accelerare la modernizzazione su larga scala di .NET, VMware mainframe e carichi di lavoro. Implementa agenti di intelligenza artificiale specializzati che automatizzano attività complesse come valutazioni, analisi del codice, refactoring, scomposizione, mappatura delle dipendenze, convalida e pianificazione della trasformazione. Questo modello dimostra come modernizzare un'applicazione mainframe e quindi distribuirla nell' AWS Transform infrastruttura utilizzando Hashicorp Terraform. AWS Queste step-by-step istruzioni consentono di trasformare CardDemo, che è un esempio di applicazione mainframe open source, da COBOL a una moderna applicazione Java.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Autorizzazioni amministrative per creare AWS risorse e distribuire applicazioni
- Terraform versione 1.5.7 o successiva, configurata
- AWS Provider per Terraform, configurato
- AWS IAM Identity Center, abilitato
- AWS Transform, abilitato
- Un utente, inserito in un'area di AWS Transform lavoro con un ruolo di collaboratore in grado di eseguire lavori di trasformazione

Limitazioni

- AWS Transform è disponibile solo in alcuni. Regioni AWS Per un elenco completo delle regioni supportate, consulta Regioni supportate per AWS Transform.
- È prevista una quota di servizio per le funzionalità di trasformazione del mainframe in AWS Transform. Per ulteriori informazioni, vedere Quotas for. AWS Transform

- Per collaborare su uno spazio di lavoro condiviso, tutti gli utenti devono essere utenti registrati della stessa istanza associata all'istanza dell' AWS Transform applicazione Web. AWS IAM Identity Center
- Il bucket AWS Transform Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) deve trovarsi nella stessa regione. Account AWS

Architecture

Il diagramma seguente mostra la end-to-end modernizzazione dell'applicazione legacy e la distribuzione in. Cloud AWS Le credenziali dell'applicazione e del database vengono Gestione dei segreti AWS archiviate e Amazon CloudWatch fornisce funzionalità di monitoraggio e registrazione.

Il diagramma mostra il flusso di lavoro seguente:

1. Tramite AWS IAM Identity Center, l'utente si autentica e accede in. AWS Transform Account AWS
2. L'utente carica il codice mainframe COBOL nel bucket Amazon S3 e avvia la trasformazione in. AWS Transform
3. AWS Transform modernizza il codice COBOL in codice Java nativo del cloud e archivia il codice modernizzato nel bucket Amazon S3.
4. Terraform crea l' AWS infrastruttura per distribuire l'applicazione modernizzata, tra cui un'Application Load Balancer, un'istanza Amazon Elastic Compute Cloud (Amazon EC2) e il database Amazon Relational Database Service (Amazon RDS). Terraform distribuisce il codice modernizzato sull'istanza Amazon EC2.
5. I file VSAM vengono caricati su Amazon EC2 e migrati da Amazon EC2 al database Amazon RDS.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel Cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente. In questo modello, le istanze del cluster di failover di SQL Server vengono installate sulle istanze Amazon EC2.
- AWS IAM Identity Centerti aiuta a gestire centralmente l'accesso Single Sign-On (SSO) alle tue applicazioni e a quelle cloud. Account AWS

- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale in. Cloud AWS
- Con Gestione dei segreti AWS puoi sostituire le credenziali nel codice,incluse le password, con una chiamata API a Secrets Manager in modo da recuperare il segreto a livello di codice.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Transformutilizza l'intelligenza artificiale agentica per aiutarti ad accelerare la modernizzazione dei carichi di lavoro legacy, come.NET, mainframe e carichi di lavoro. VMware

Altri strumenti

- Apache Maven è uno strumento open source per la gestione di progetti software e l'automazione dello sviluppo per progetti Java.
- Apache Tomcat è un contenitore Servlet open source e un server web per codice Java.
- HashiCorp Terraform è uno strumento Infrastructure as Code (IaC) che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud.
- Spring Boot è un framework open source basato su Spring Framework in Java.

Deposito di codice

Il codice per questo pattern è disponibile nel repository GitHub Mainframe Transformation E2E. Questo modello utilizza l'applicazione AWS CardDemomainframe open source come applicazione di esempio.

Best practice

- Assegna la piena proprietà del codice e delle risorse destinate alla migrazione.
- Sviluppa e testa un proof of concept prima di passare a una migrazione completa.
- Garantisci l'impegno di tutte le parti interessate.
- Stabilisci canali di comunicazione chiari.
- Definisci e documenta i requisiti minimi di prodotto (MVP).
- Stabilisci criteri di successo chiari.

Epiche

Prepara e carica il codice dell'applicazione mainframe

Operazione	Description	Competenze richieste
Creare un bucket.	Crea un bucket Amazon S3 nella stessa regione in cui AWS Transform è Account AWS abilitato. Questo bucket viene utilizzato per archiviar e il codice dell'applicazione mainframe, i dati e gli script aggiuntivi necessari per creare ed eseguire l'applica zione. AWS Transform utilizza questo bucket per archiviare il codice rifattorizzato e altri file associati alla trasforma zione. Per istruzioni, consulta Creazione di un bucket nella documentazione di Amazon S3.	AWS generale, amministratore AWS
Imposta le autorizzazioni CORS per il bucket.	Quando configuri il bucket per AWS Transform l'accesso, devi configurare la condivisi one delle risorse tra le origini (CORS) per il bucket. Se non è configurato correttamente, potresti non essere in grado di utilizzare le funzionalità di visualizzazione in linea o di confronto dei file di. AWS Transform Per istruzioni su come configurare CORS per un bucket, consulta Usare	AWS generale, amministratore AWS

Operazione	Description	Competenze richieste
	la condivisione di risorse tra origini diverse nel bucket Amazon S3. Per la policy, consulta le autorizzazioni CORS del bucket S3 nella documentazione. AWS Transform	
Preparare il codice applicativo mainframe di esempio.	1. Immettete il seguente comando per clonare il CardDemorepository sulla workstation locale: `git clone https://github.com/aws-samples/aws-mainframe-modernization-carddemo.git` 2. Comprimi la `aws-mainframe-modernization-carddemo` cartella in un file ZIP. 3. Carica il file ZIP nel bucket Amazon S3 che hai creato. Per istruzioni, consulta Caricamento di oggetti nella documentazione di Amazon S3.	General AWS, sviluppatore di app

Trasforma l'applicazione mainframe

Operazione	Description	Competenze richieste
Configura il AWS Transform lavoro.	1. Accedi all'applicazione AWS Transform web effettuando il login con le tue credenziali. 2. Crea un nuovo spazio di lavoro seguendo le istruzioni in Configurazione dell'area di lavoro nella documentazione. AWS Transform 3. Nella pagina di destinazione del tuo spazio di lavoro, scegli Chiedi AWS Transform per creare un lavoro. 4. Quindi, scegli Mainframe Modernization come tipo di lavoro. 5. Nella finestra della chat, inserisci Transform code to Java. 6. Controlla il tipo di lavoro, il nome e l'obiettivo suggeriti. Per confermare, inserisci Sì. 7. Scegli Crea processo.	Sviluppatore dell'app, proprietario dell'app
Configura un connettore.	1. Configura un connettore con il bucket Amazon S3 che hai creato. Per istruzioni, consulta Configurare un connettore nella AWS Transform documenta	Sviluppatore di app, proprietario dell'app

Operazione	Description	Competenze richieste
	zione. Quando richiesto, inserisci il percorso del file `aws-mainframe-modernization-carddemo.zip` nel bucket Amazon S3. 2. Attendi il completamento del processo.	
Trasforma il codice.	1. Esamina i risultati dell'analisi del codice secondo le istruzioni in Analizza il codice nella AWS Transform documentazione. 2. Rifattorizza il codice del mainframe in base alle istruzioni contenute nel codice Refactor nella documentazione. AWS Transform Per l'applicazione CardDemo mainframe di esempio, puoi accettare le impostazioni predefinite. 3. Attendi il completamento del refactoring. 4. Scegli Visualizza risultati per visualizzare il percorso del codice refactorizzato nel bucket Amazon S3. Prendi nota del percorso di questo file. Sarà necessario in seguito.	Sviluppatore dell'app, proprietario dell'app

Implementa l'infrastruttura tramite Terraform

Operazione	Description	Competenze richieste

Aggiorna i modelli.	1. Immettete il seguente comando per clonare il repository Mainframe Transformation E2E sulla workstation locale:
	`git clone https://github.com/aws-samples/sample-mainframe-transformation-e2e.git`
	2. Immettete il seguente comando per recuperare il vostro attuale indirizzo IP pubblico:
	`curl checkip.amazonaws.com`
	3. Immettete il seguente comando per accedere alla directory infra:
	`cd mainframe-transformation-e2e/infra`
	4. Apri il file `variables.tf` .
	5. Sostituisci `YOUR_IP_ADDRESS_HERE` con il tuo indirizzo IP.
	6. Se disponi di una zona ospitata pubblica, procedi come segue:
		AWS generale, amministratore AWS

Operazione	Description	Competenze richieste

	a. `hosted_zone_name` Sostituiscila con il nome della tua zona ospitata.
	b. Imposta `hosted_zone_enabled` su `true`.
	
	Se non disponi di una zona ospitata pubblica, procedi come segue:
	a. Inserisci i seguenti comandi per generare un certificato autofirmato:
	
	`openssl genrsa 2048 > my-private-key.pem`
	`openssl req -new -x509 -nodes -sha256 -days 365 -key my-private-key.pem -outform PEM -out my-certificate.pem`
	
	b. Immettete il seguente comando per importare il certificato in AWS Certificate Manager (ACM):
	
	`aws acm import-certificate \`
	`--certificate fileb://my-certificate.pem \`	

Operazione	Description	Competenze richieste
	`--private-key fileb://my-private-key.pem` L'output di questo comando include l'Amazon Resource Name (ARN) del certificato importato. c. Sostituisci `self_signed_cert_arn` con l'ARN del tuo certificato. d. Imposta `hosted_zone_enabled` su `false`. 7. Passa `aws_region` alla regione di destinazione. Il valore predefinito è `us-east-1` . 8. Salvare e chiudere il file `variables.tf` . **Note** Per ambienti di produzione o simili, configura componenti di sicurezza aggiuntivi. Ad esempio, abilita AWS WAF le protezioni per il tuo Application Load Balancer.	

Operazione	Description	Competenze richieste

Implementa l'infrastruttura.	1. Inserisci il seguente comando per inizializzare Terraform:
	`terraform init`
	2. Immettere il seguente comando per generare un piano di esecuzione:
	`terraform plan`
	3. Rivedi il piano e convalida le risorse e i component i dell'infrastruttura che verranno creati.
	4. Immettere il seguente comando per distribuire l'infrastruttura:
	`terraform apply`
	5. Quando richiesto, inserisci yes per confermare la distribuzione.
	6. Attendi il completamento della distribuzione.
		Terraform

Installa e configura Apache Tomcat sull'istanza Amazon EC2

Operazione	Description	Competenze richieste
Installa il software richiesto.	1. Connect alla tua istanza Amazon EC2 utilizzando.	Sviluppatore di app, tecnico addetto alla migrazione

Operazione	Description	Competenze richieste

	AWS Systems Manager Session Manager
	2. Inserisci il seguente comando per passare all'utente root:
	`sudo su -`
	3. Immettere il comando seguente per andare alla directory `scripts`:
	`cd /opt/scripts`
	4. Esamina lo script install_software.sh. Questo script installa Java 17, Apache Maven e Apache Tomcat 10.0.23.
	5. Aggiorna gli script secondo necessità per il tuo caso d'uso.
	6. Immettete il seguente comando per rendere eseguibile lo script:
	`chmod +x install_software.sh`
	7. Immettete il seguente comando per eseguire lo script:
	`./install_software.sh`	

Operazione	Description	Competenze richieste

Verifica l'installazione del software.	1. Immettere il seguente comando per avviare il server Tomcat:
	`/opt/tomcat/apache-tomcat-10.0.23/bin/startup.sh`
	2. Immettere il comando seguente per verificare la risposta del server Web:
	`curl http://localhost:8080`
	L'output dovrebbe confermare che Tomcat sta servendo una pagina HTML.
		Sviluppatore di app, tecnico addetto alla migrazione

Compila e impacchetta il codice applicativo modernizzato

Operazione	Description	Competenze richieste
Scarica ed estrai il codice generato.	1. Immettete il seguente comando per rendere eseguibile lo script download_and_extract.sh. Questo script scarica il codice refattorizzato e la libreria di runtime Gapwalk archiviata nel bucket Amazon S3:	Sviluppatore di app, tecnico addetto alla migrazione

Operazione	Description	Competenze richieste

	`chmod +x /opt/scripts/download_and_extract.sh`
	2. Inserisci il comando seguente per eseguire lo script . Sostituisci `<file_path>` con il percorso del file generated .zip nel tuo bucket Amazon S3:
	`./download_and_extract.sh <file_path>`
	Note
	Il percorso del file è in genere. `s3://<bucket-name>/transform_output/<aws_transform_job_id>/code transformation/generated.zip`
	3. Immettete il seguente comando per navigare all'interno della `shared` cartella:	

Operazione	Description	Competenze richieste

	`cd /opt/runtime/velocity/shared`
	4. Immettete il seguente comando per copiare lo deploy-velocity-runtimescript.sh:
	`cp /opt/scripts/deploy-velocity-runtime.sh .`
	5. Immettete il comando seguente per rendere eseguibile lo script copiato:
	`chmod +x deploy-velocity-runtime.sh`
	6. Inserisci il comando seguente per eseguire lo script . Questo script copia tutte le dipendenze WAR (Web Application Archive) richieste presenti nei file Project Object Model (POM) nella cartella del repository:
	`./deploy-velocity-runtime.sh`
	7. Verifica la corretta esecuzione controllando che non vi siano errori e che le dipendenze WAR	

Operazione	Description	Competenze richieste
	richieste siano installate nel repository Maven locale.	

Operazione	Description	Competenze richieste

Crea l'applicazione modernizzata.	1. Immettete il seguente comando per accedere alla directory del `app-pom` progetto:
	`cd /opt/codebase/app-pom/`
	2. Inserisci il seguente comando per installare Maven:
	`mvn clean install`
	3. Attendi il completamento dell'installazione e della compilazione.
	Note
	Quando si esegue questo comando per l' CardDemo applicazione, è possibile che vengano visualizzati messaggi di avviso relativi al `app-web` progetto. Puoi tranquillamente ignorare questi avvisi.
	4. Dopo aver completato con successo la build, conferma la presenza di `app-servi`
		Sviluppatore di app, tecnico addetto alla migrazione

Operazione	Description	Competenze richieste
	`ce/target/app-service-1.0.0.war eapp-web/target/app-web-1.0.0.war .` **Important** Non riavviare il server Tomcat in questa fase. Ciò comporterebbe errori dovuti all'assenza dei database richiesti. È necessario configurare il database prima di riavviare il server.	

Esegui la migrazione del database

Operazione	Description	Competenze richieste
Creare il database e gli schemi JICS.	1. Immettere il seguente comando per rinominare la cartella in `csd commands`. `csd_commands` Questo rimuove gli spazi dal nome della cartella: `mv /opt/codebase/extra/csd\ commands/ /opt/codebase/extra/csd_commands`	Sviluppatore di app, tecnico addetto alla migrazione

Operazione	Description	Competenze richieste

	2. Immettete il seguente comando per navigare nella `scripts` directory:
	`cd /opt/scripts`
	3. Immettete il seguente comando per rendere eseguibile lo script di migrazione del database:
	`chmod +x database_migration_setup.sh`
	4. Immettete i seguenti comandi per configurare le seguenti variabili come parametri:
	`RDS_ENDPOINT=<database_endpoint>`
	`SECRET_NAME=<secret_name>`
	`JICS_SQL_SCRIPT_DIR=/opt/runtime/velocity/jics/sql/jics.sql`
	`INIT_JICS_SQL_SCRIPT_DIR=/opt/codebase/extra/csd_commands/sql/aws-mainframe-modernization-carddemo-main/app/csd/initJics.sql`
	Dove:
	• `<database_endpoint>` è l'endpoint del	

Operazione	Description	Competenze richieste

	database Amazon RDS che hai distribuito tramite Terraform.
	• `<secret_name>` è il nome del Gestione dei segreti AWS segreto che hai distribuito tramite Terraform.
	5. Immettere il seguente comando per eseguire lo script di migrazione del database:
	`./database_migration_setup.sh $RDS_ENDPOINT $SECRET_NAME $JICS_SQL_SCRIPT_DIR $INIT_JICS_SQL_SCRIPT_DIR`
	6. Inserisci il seguente comando per connetterti al database dalla tua istanza Amazon EC2:
	`psql -h <Your Amazon RDS Endpoint> -U foo -p 5432 postgres`
	7. Quando richiesto, inserisci le credenziali del database.	

Operazione	Description	Competenze richieste

Convalida la creazione del database.	1. Immettete il seguente comando per visualizzare tutti i database:
	`\l`
	2. Immettete il seguente comando per passare al `jics` database:
	`\c jics`
	3. Immettete il comando seguente per rivedere un elenco delle tabelle create:
	`\dt`
		Sviluppatore di app, tecnico addetto alla migrazione

Operazione	Description	Competenze richieste

Migra i dati nel database JICS.	1. Immettete il seguente comando per rendere eseguibile lo script execute_listcat_sql.sh:
	`chmod +x execute_listcat_sql.sh`
	2. Immettete il comando seguente per configurare la `PATH_TO_LISTCAT_SQL_FILES` variabile, che è la directory che contiene i file SQL LISTCAT:
	`PATH_TO_LISTCAT_SQL_FILES=/opt/codebase/extra/listcat/sql/cluster/aws-mainframe-modernization-carddemo-main/app/catlg`
	3. Assicurati che le `PATH_TO_LISTCAT_SQL_FILES` variabili `RDS_ENDPOINT SECRET_NAME` , e siano impostate correttamente in base alle istruzioni precedenti.
	4. Immettete il comando seguente per eseguire lo script execute_listcat_sql.sh:
		Sviluppatore di app, tecnico addetto alla migrazione

Operazione	Description	Competenze richieste
	`./execute_listcat_sql.sh $RDS_ENDPOINT $SECRET_NAME $PATH_TO_LISTCAT_SQL_FILES` Questo script aggiorna le proprietà del file VSAM nel database JICS ed esegue le query necessarie per modificare il database.	

Installa l'applicazione modernizzata

Operazione	Description	Competenze richieste

Installa l'applicazione modernizzata sull'istanza Amazon EC2.	1. Inserisci il seguente comando per rendere eseguibile lo script application_installer.sh:
	
	`chmod +x /opt/scripts/application_installer.sh`
	
	2. Immettete i seguenti comandi per configurare le seguenti variabili come parametri:
	
	`RDS_ENDPOINT=<database_endpoint>`
	`SECRET_NAME=<secret_name>`
	`AIX_JSON_FILE_PATH=/opt/codebase/extra/`
		Sviluppatore di app, architetto cloud

Operazione	Description	Competenze richieste
	csd_commands/json/aws-mainframe-modernization-carddemo-main/jicsFileAix.json LISTCAT_JSON_FILES_DIR=/opt/codebase/extra/listcat/json/cluster/default/aws-mainframe-modernization-carddemo-main/app/catlg S3_PATH_FOR_EBCDIC_DATA_FILES=s3://<bucket_name>/transform-output/<job_id>/inputs/aws-mainframe-modernization-carddemo-main/app/data/EBCDIC Dove: • <database_endpoint> è l'endpoint del database Amazon RDS che hai distribuito tramite Terraform. • <secret_name> è il nome del Gestione dei segreti AWS segreto che hai distribuito tramite Terraform. • <bucket_name> è il nome del bucket Amazon S3 che contiene l'applicazione modernizzata.	

Operazione	Description	Competenze richieste

	• `<job_id>` è l'ID del lavoro. AWS Transform
	3. Immettete il comando seguente per eseguire lo script application_installer.sh:
	`./application_installer.sh $RDS_ENDPOINT $SECRET_NAME $AIX_JSON_FILE_PATH $LISTCAT_JSON_FILES_DIR $S3_PATH_FOR_EBCDIC_DATA_FILES`
	4. Nella `/opt/tomcat/apache-tomcat-10.0.23/workingdir/config` cartella, nel file application-utility-pgm.yml, modificate il parametro di codifica nel modo seguente:
	`encoding : CP1047`
	Note
	Quando esegui il refactoring AWS Transform automatico delle applicazioni utilizzando AWS Blu Age, configuri l'applica	

Operazione	Description	Competenze richieste
	zione e il relativo ambiente di runtime tramite file YAML. Ad esempio, è possibile configura re la registrazione nel file applicati on-main.yml per l'applicazione. Per ulteriori informazi oni sulle proprietà disponibili, vedere Abilitare le proprietà per Blu Age Runtime. AWS	

Operazione	Description	Competenze richieste

Riavvia il server Tomcat.	1. Immettete il seguente comando per navigare nella directory di lavoro:
	`cd /opt/tomcat/apache-tomcat-10.0.23/workingdir`
	2. Immettete i seguenti comandi per arrestare e avviare il server Tomcat:
	`/opt/tomcat/apache-tomcat-10.0.23/bin/shutdown.sh`
	`/opt/tomcat/apache-tomcat-10.0.23/bin/startup.sh`
	3. Immettere il seguente comando per monitorare i log di avvio del servizio Tomcat:
	`tail -f /opt/tomcat/apache-tomcat-10.0.23/logs/catalina.out`
		Sviluppatore di app, architetto cloud

Operazione	Description	Competenze richieste
Esegui la migrazione del set di dati VSAM.	1. Aprire la console di Amazon EC2. 2. Nel riquadro di navigazione selezionare Load Balancers (Sistemi di bilanciamento del carico). 3. Scegli il load balancer creato tramite Terraform. 4. Individua il nome DNS (Domain Name System) del tuo Application Load Balancer, ad esempio. `application-load-balancer-<id>.<region>.elb.amazonaws.com` 5. Nel browser, accedi a `http://<dns_name>/gapwalk-application/scripts/data-load` , `<dns_name>` dov'è il nome DSN dell'Application Load Balancer. Questo avvia lo script di caricamento dei dati. 6. Attendi il completamento dello script. Al termine, il browser mostrerà DONE. 7. Nell'istanza Amazon EC2, apri un nuovo terminale. 8. Inserisci il seguente comando per connetterti al database Amazon RDS,	Sviluppatore di app, tecnico addetto alla migrazione

Operazione	Description	Competenze richieste

	sostituendolo <database_endpoint> con il tuo valore:
	`psql -h <database_endpoint> -U foo -p 5432 postgres`
	9. Quando richiesto, inserisci le tue credenziali per connetterti al database.
	10. Immettete il seguente comando per visualizzare tutti i database:
	`\l`
	11. Immettete il seguente comando per passare al `bluesam` database:
	`\c bluesam`
	12. Immettete il comando seguente per rivedere un elenco delle tabelle create:
	`\dt`
	13. Immettete il comando seguente per convalidare il caricamento dei dati:
	`SELECT * FROM public.aws_m2_carddemo_usrsec_vsam_ksds;`	

Operazione	Description	Competenze richieste
	L'output dovrebbe mostrare 10 record restituiti.	
Aggiorna i parametri negli script Groovy.	1. Immettete il seguente comando per navigare nella directory degli script: `cd /opt/tomcat/apache-tomcat-10.0.23/webapps/workingdir/scripts` 2. In tutti i file Groovy che contengono riferimenti a file flat, aggiorna le seguenti configurazioni di file: • Percorso del file locale: il percorso del file flat nella directory locale dell'istanza Amazon EC2 • Tipo di file system: il file system che contiene i file flat • Dimensione del record: la dimensione del file flat Per riferimento, vedete lo script di esempio dusrSecj.jcl.Groovy nel repository del codice. 3. Salvate e chiudete i file.	Sviluppatore di app

Eseguire il test dell'applicazione

Operazione	Description	Competenze richieste
Prova l'applicazione modernizzata.	1. Accedere all'applicazione online tramite Application Load Balancer (`http://<your-load-balancer-dns>`) o tramite la zona ospitata (`https://myhostedzone.dev/`). 2. Per l'ID della transazione, immettere`CC00`. 3. Per il nome utente, immettere`USER0001`. 4. Per la password, immettere `PASSWORD`. Dopo aver effettuato correttamente l'accesso, viene visualizzato il menu principale.	Sviluppatore di app, tecnico di test
Verifica gli script batch.	1. Accedi all'interfaccia degli script tramite Application Load Balancer `http://<your-load-balancer-dns>/gapwalk-application/scripts` () o tramite la zona `https://myhostedzone.dev/gapwalk-application/scripts` ospitata (). 2. Scegliete uno script da eseguire, ad esempio lo script DUSRSecj.jcl.groovy.	Sviluppatore di app, tecnico di test

Operazione	Description	Competenze richieste
	3. Verificate che lo script venga eseguito correttamente. Di seguito è riportato un esempio di output dopo un'esecuzione riuscita. `{ "exitCode": 0, "stepName": "STEP03", "program": "IDCAMS", "status": "Succeeded" }`	

Eliminazione

Operazione	Description	Competenze richieste
Preparati a eliminare l'infrastruttura.	1. Inserisci il seguente comando per rimuovere la protezione da eliminazione dall'istanza Amazon RDS: `aws rds modify-db-instance \ --db-instance-identifier <your-db-instance-name> \ --no-deletion-protection \ --apply-immediately` 2. Immettete il seguente comando per rimuovere la protezione da eliminazione dall'Application Load Balancer:	Informazioni generali su AWS

Operazione	Description	Competenze richieste

	`aws elbv2 modify-load-balancer-attributes \`
	`  --load-balancer-arn <your-load-balancer-arn> \`
	`  --attributes Key=deletion_protection.enabled,Value=false`	

	3. Inserisci i seguenti comandi per eliminare il contenuto dei bucket Amazon S3:
	`ACCOUNT_NUMBER=$(aws sts get-caller-identity --query Account --output text)`
	`aws s3 rm s3://mf-carddemo-$ACCOUNT_NUMBER --recursive`
	`aws s3 rm s3://mf-carddemo-logs-$ACCOUNT_NUMBER --recursive`	

Operazione	Description	Competenze richieste

Eliminare l'infrastruttura.	**Important**
	Questi passaggi elimineranno definitiv amente le tue risorse. Assicurati di aver eseguito il backup di tutti i dati importanti prima di procedere.
	
	1. Immettete il seguente comando per navigare all'interno della `infra` cartella:
	`cd /mainframe-transformation-e2e/infra`
	2. Immettere il seguente comando per eliminare l'infrastruttura:
	`terraform destroy --auto-approve`
		Informazioni generali su AWS

Risoluzione dei problemi

Problema	Soluzione
Autenticazione Terraform	1. Assicurati che le AWS credenziali siano configurate correttamente. 2. Verifica di aver selezionato il AWS profilo corretto.

Problema	Soluzione
	3. Conferma di disporre delle autorizzazioni necessarie.
Errori relativi a Tomcat	1. Controlla catalina.out per eventuali eccezioni . `/opt/tomcat/apache-tomcat-10.0.23/logs` 2. Immetti il seguente comando per cambiare la proprietà della cartella Tomcat all'utente Tomcat: `chown -R tomcat:tomcat /opt/tomcat/*`
Il nome dell'URL non viene caricato	Assicurati che il gruppo di sicurezza Application Load Balancer abbia il tuo indirizzo IP nella regola in entrata come origine.
Problema di autenticazione nel registro di Tomcat	Verifica che la password segreta del database in Gestione dei segreti AWS e la password in server.xml corrispondano.

Risorse correlate

AWS Guida prescrittiva

- Modernizza l'applicazione CardDemo mainframe utilizzando AWS Transform

AWS documentazione del servizio

- AWS Console di amministrazione Blue Age Blusam
- Requisiti di configurazione dell'infrastruttura per AWS Blu Age Runtime (non gestito)
- Onboarding AWS di Blu Age Runtime
- Modernizzazione delle applicazioni mainframe
- Configura la configurazione per AWS Blu Age Runtime

AWS post sul blog

- Accelera il tuo percorso di modernizzazione del mainframe utilizzando agenti AI con AWS Transform

Modernizza la gestione dell'output del mainframe AWS utilizzando Rocket Enterprise Server e LRS X PageCenter

Shubham Roy, Amazon Web Services

Abraham Rondon, Micro Focus

Guy Tucker, Levi, Ray and Shoup Inc

Riepilogo

Modernizzando la gestione dell'output del mainframe, puoi ottenere risparmi sui costi, mitigare il debito tecnico legato alla manutenzione dei sistemi legacy e migliorare la resilienza e l'agilità attraverso tecnologie native per il cloud di Amazon Web DevOps Services (AWS). Questo modello mostra come modernizzare i carichi di lavoro di gestione dell'output mainframe critici per l'azienda sul cloud AWS. Il modello utilizza Rocket Enterprise Server come runtime per un'applicazione mainframe modernizzata, con Levi, Ray & Shoup, Inc. (LRS) VPSX/MFI (Micro Focus Interface) come server di stampa e LRS X come server di archiviazione. PageCenter LRS PageCenter X offre soluzioni di gestione dell'output per la visualizzazione, l'indicizzazione, la ricerca, l'archiviazione e la protezione dell'accesso agli output aziendali.

Il modello si basa sull'approccio di modernizzazione del mainframe replatform. Le applicazioni mainframe vengono migrate da AWS Mainframe Modernization su Amazon Elastic Compute Cloud (Amazon). EC2 I carichi di lavoro di gestione dell'output mainframe vengono migrati su Amazon e un database mainframe EC2, come IBM Db2 for) funziona con z/OS, is migrated to Amazon Relational Database Service (Amazon RDS). The LRS Directory Integration Server (LRS/DIS AWS Directory Service per Microsoft Active Directory per l'autenticazione e l'autorizzazione del flusso di lavoro di gestione dell'output.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un carico di lavoro di gestione dell'output su mainframe.
- Conoscenze di base su come ricostruire e fornire un'applicazione mainframe eseguibile su Rocket Enterprise Server. Per ulteriori informazioni, consulta la scheda tecnica di Rocket Enterprise Server nella documentazione di Rocket Software.

- Conoscenza di base delle soluzioni e dei concetti di stampa su cloud di LRS. Per ulteriori informazioni, consultate Output Modernization nella documentazione di LRS.
- Software e licenza Rocket Enterprise Server. Per ulteriori informazioni, contattate Rocket Software.
- VPSX/MFI, LRS PageCenterX, LRS/Queue, and LRS/DISSoftware e licenze LRS. Per ulteriori informazioni, contattate LRS. È necessario fornire i nomi host delle EC2 istanze in cui verranno installati i prodotti LRS.

Nota: per ulteriori informazioni sulle considerazioni sulla configurazione per i carichi di lavoro di gestione dell'output del mainframe, vedere Considerazioni nella sezione Informazioni aggiuntive di questo modello.

Versioni del prodotto

- Rocket Enterprise Server 10.0
- LRS VPSX/MFI
- LRS X PageCenter V1R3 o versione successiva

Architecture

Stack tecnologico di origine

- Sistema operativo: IBM z/OS
- Linguaggio di programmazione: Common Business-Oriented Language (COBOL), Job Control Language (JCL) e Customer Information Control System (CICS)
- Database: IBM Db2 for z/OS, database IBM Information Management System (IMS) e Virtual Storage Access Method (VSAM)
- Sicurezza: Resource Access Control Facility (RACF), CA Top Secret for z/OS e Access Control Facility 2 () ACF2
- Soluzioni di stampa e archiviazione: prodotti di z/OS output e stampa mainframe IBM (IBM Infoprint Server for z/OS, LRS e CA Deliver) e soluzioni di archiviazione (CA Deliver, ASG Mobius o CA Bundle)

Architettura di origine

Il diagramma seguente mostra una tipica architettura allo stato attuale per un carico di lavoro di gestione dell'output su mainframe.

Il diagramma mostra il flusso di lavoro seguente:

1. Gli utenti eseguono transazioni commerciali su un sistema di coinvolgimento (SoE) basato su un'applicazione IBM CICS scritta in COBOL.
2. Il SoE richiama il servizio mainframe, che registra i dati delle transazioni commerciali in un database system-of-records (SoR) come IBM Db2 for z/OS.
3. Il SoR conserva i dati aziendali del SoE.
4. Il batch job scheduler avvia un processo batch per generare l'output di stampa.
5. Il processo batch estrae i dati dal database. Formatta i dati in base ai requisiti aziendali, quindi genera risultati aziendali come dichiarazioni di fatturazione, carte d'identità o dichiarazioni di prestito. Infine, il processo in batch indirizza l'output alla gestione dell'output per la formattazione, la pubblicazione e l'archiviazione dell'output in base ai requisiti aziendali.
6. La gestione dell'output riceve l'output dal processo batch. La gestione dell'output indicizza, organizza e pubblica l'output in una destinazione specificata nel sistema di gestione dell'output, ad esempio le soluzioni LRS PageCenter X (come illustrato in questo modello) o CA View.
7. Gli utenti possono visualizzare, cercare e recuperare l'output.

Stack tecnologico Target

- Sistema operativo: Windows Server in esecuzione su Amazon EC2
- Elaborazione — Amazon EC2
- Storage: Amazon Elastic Block Store (Amazon EBS) e FSx Amazon per Windows File Server
- Linguaggio di programmazione: COBOL, JCL e CICS
- Banca dati — Amazon RDS
- Sicurezza: AWS Managed Microsoft AD
- Stampa e archiviazione: soluzione di stampa (VPSX) e archiviazione (PageCenterX) LRS su AWS
- Ambiente di runtime mainframe: Rocket Enterprise Server

Architettura di destinazione

Il diagramma seguente mostra un'architettura per un carico di lavoro di gestione dell'output mainframe distribuito nel cloud AWS.

Il diagramma mostra il flusso di lavoro seguente:

1. Il batch job scheduler avvia un processo batch per creare output, ad esempio estratti conto di fatturazione, carte d'identità o rendiconti di prestito.
2. Il processo batch del mainframe (riplatformato su Amazon EC2) utilizza il runtime di Rocket Enterprise Server per estrarre i dati dal database dell'applicazione, applicare la logica di business ai dati e formattare i dati. Quindi invia i dati a una destinazione di output utilizzando il modulo di uscita della stampante Rocket Software (OpenText documentazione Micro Focus).
3. Il database dell'applicazione (un SoR eseguito su Amazon RDS) mantiene i dati per l'output di stampa.
4. La soluzione di VPSX/MFI stampa LRS è implementata su Amazon e EC2 i suoi dati operativi sono archiviati in Amazon EBS. LRS VPSX/MFI utilizza l'agente di TCP/IP-based LRS/Queue trasmissione per raccogliere i dati di output tramite l'API JES Print Exit di Rocket Software.

   LRS VPSX/MFI esegue la preelaborazione dei dati, ad esempio la traduzione da EBCDIC a ASCII. Svolge anche attività più complesse, inclusa la conversione di flussi di dati esclusivi del mainframe come IBM Advanced Function Presentation (AFP) e Xerox Line Conditioned Data Stream (LCDS) in flussi di dati di visualizzazione e stampa più comuni come Printer Command Language (PCL) e PDF.

   Durante la finestra di manutenzione di LRS PageCenter X, LRS mantiene la coda di output e funge da backup per la coda di output VPSX/MFI . LRS VPSX/MFI si connette e invia l'output a LRS X utilizzando il protocollo PageCenter. LRS/Queue LRS/Queue effettua uno scambio di informazioni sullo stato di preparazione e sul completamento dei lavori per garantire che il trasferimento dei dati avvenga.

   Note:

   Per ulteriori informazioni sui dati di stampa trasferiti da Rocket Software Print Exit a LRS/Queue e sui meccanismi batch mainframe VPSX/MFI supportati da LRS, consultate Print Data Capture nella sezione Informazioni aggiuntive.

LRS VPSX/MFI può eseguire controlli di integrità a livello di parco stampanti. Per ulteriori informazioni, vedete Controlli dello stato del parco stampanti nella sezione Informazioni aggiuntive di questo modello.

5. La soluzione di gestione dell'output LRS PageCenter X è distribuita su Amazon EC2 e i suoi dati operativi sono archiviati in Amazon FSx for Windows File Server. LRS PageCenter X fornisce un sistema centrale di gestione dei report di tutti i file importati in LRS PageCenter X e consente a tutti gli utenti di accedervi. Gli utenti possono visualizzare contenuti di file specifici o eseguire ricerche su più file per trovare criteri corrispondenti.

   Il LRS/NetX componente è un server di applicazioni Web multithread che fornisce un ambiente di runtime comune per l'applicazione LRS PageCenter X e altre applicazioni LRS. Il componente LRS/Web Connect è installato sul server Web e fornisce un connettore dal server Web al server delle applicazioni LRS/NetX Web.
6. LRS PageCenter X fornisce l'archiviazione per gli oggetti del file system. I dati operativi di LRS PageCenter X sono archiviati in Amazon FSx for Windows File Server.
7. L'autenticazione e l'autorizzazione della gestione dell'output vengono eseguite da AWS Managed Microsoft AD con LRS/DIS.

> **Note**
>
> La soluzione di destinazione in genere non richiede modifiche alle applicazioni per adattarsi ai linguaggi di formattazione del mainframe, come IBM AFP o Xerox LCDS.

Architettura dell'infrastruttura AWS

Il diagramma seguente mostra un'architettura di infrastruttura AWS altamente disponibile e sicura per un carico di lavoro di gestione dell'output mainframe.

Il diagramma mostra il flusso di lavoro seguente:

1. Lo scheduler batch avvia il processo batch e viene distribuito su Amazon EC2 su più zone di disponibilità per l'alta disponibilità (HA).

> **Note**
>
> Questo modello non copre l'implementazione dello scheduler in batch. Per ulteriori informazioni sull'implementazione, consultate la documentazione del fornitore del software relativa allo scheduler in uso.

2. Il processo batch del mainframe (scritto in un linguaggio di programmazione come JCL o COBOL) utilizza la logica aziendale principale per elaborare e generare output di stampa, come estratti conto di fatturazione, carte d'identità e dichiarazioni di prestito. Il processo batch viene distribuito su Amazon EC2 in due zone di disponibilità per HA. Utilizza l'API Print Exit di Rocket Software per indirizzare l'output di stampa a LRS VPSX/MFI per la preelaborazione dei dati.
3. Il server di VPSX/MFI stampa LRS è distribuito su Amazon EC2 in due zone di disponibilità per HA (coppia ridondante attiva-standby). Utilizza Amazon EBS come archivio dati operativo. Il Network Load Balancer esegue un controllo dello stato delle istanze LRS VPSX/MFI EC2 . Se un'istanza attiva non è integra, il load balancer indirizza il traffico verso le istanze hot standby nell'altra zona di disponibilità. Le richieste di stampa vengono mantenute nella LRS Job Queue localmente in ciascuna istanza. EC2 In caso di errore, è necessario riavviare un'istanza fallita prima che i servizi LRS possano riprendere l'elaborazione della richiesta di stampa.

> **Note**
>
> LRS VPSX/MFI può anche eseguire controlli di integrità a livello di parco stampanti. Per ulteriori informazioni, vedete Controlli dello stato del parco stampanti nella sezione Informazioni aggiuntive di questo modello.

4. La gestione dell'output di LRS PageCenter X è implementata EC2 su Amazon in due zone di disponibilità per HA (coppia ridondante attiva-standby). Utilizza Amazon FSx for Windows File Server come archivio dati operativo. Se un'istanza attiva non è integra, il load balancer esegue un controllo dello stato delle istanze LRS PageCenter X e indirizza il traffico verso EC2 le istanze di standby nell'altra zona di disponibilità.
5. Un Network Load Balancer fornisce un nome DNS per integrare il server LRS con LRS VPSX/MFI X. PageCenter

> **Note**
>
> LRS PageCenter X supporta un sistema di bilanciamento del carico di livello 4.

6. LRS PageCenter X utilizza Amazon FSx for Windows File Server come archivio dati operativo distribuito su due zone di disponibilità per HA. LRS PageCenter X comprende solo i file che si trovano nella condivisione di file, non in un database esterno.
7. AWS Managed Microsoft AD viene utilizzato per LRS/DIS eseguire l'autenticazione e l'autorizzazione del flusso di lavoro di gestione dell'output. Per ulteriori informazioni, consulta Autenticazione e autorizzazione dell'output di stampa nella sezione Informazioni aggiuntive.

Tools (Strumenti)

Servizi AWS

- AWS Directory Service per Microsoft Active Directory consente ai carichi di lavoro compatibili con le directory e alle risorse AWS di utilizzare Microsoft Active Directory nel cloud AWS.
- Amazon Elastic Block Store (Amazon EBS) fornisce volumi di storage a livello di blocco da utilizzare con istanze Amazon Elastic Compute Cloud (Amazon). EC2
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Elastic Load Balancing (ELB) distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni. Ad esempio, puoi distribuire il traffico tra EC2 istanze Amazon, contenitori e indirizzi IP in una o più zone di disponibilità. Questo modello utilizza un Network Load Balancer.
- Amazon FSx fornisce file system che supportano i protocolli di connettività standard del settore e offrono disponibilità e replica elevate in tutte le regioni AWS. Questo modello utilizza Amazon FSx per Windows File Server.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS.

Altri strumenti

- Il software LRS PageCenter X offre una soluzione scalabile per la gestione dei contenuti di documenti e report che aiuta gli utenti a ottenere il massimo valore dalle informazioni attraverso l'indicizzazione automatizzata, la crittografia e le funzionalità di ricerca avanzate.
- LRS VPSX/MFI (Micro Focus Interface), sviluppata congiuntamente da LRS e Rocket Software, acquisisce l'output da una bobina JES di Rocket Software e lo consegna in modo affidabile a una destinazione di stampa specificata.

- LRS/Queue is a transmission agent that’s TCP/IP based. LRS VPSX/MFI uses LRS/Queueper raccogliere o acquisire dati di stampa tramite l'interfaccia di programmazione JES Print Exit di Rocket Software.
- LRS Directory Integration Server (LRS/DIS) viene utilizzato per l'autenticazione e l'autorizzazione durante il flusso di lavoro di stampa.
- Rocket Enterprise Server è un ambiente di distribuzione delle applicazioni per applicazioni mainframe. Fornisce l'ambiente di runtime per le applicazioni mainframe che vengono migrate o create utilizzando qualsiasi versione di Rocket Enterprise Developer.

Epiche

Configura il runtime Rocket e distribuisci un'applicazione batch mainframe

Operazione	Description	Competenze richieste
Configura il runtime e distribuisci un'applicazione demo.	Per configurare Rocket Enterprise Server su Amazon EC2 e distribuire l'applicazione BankDemo dimostrativa Rocket Software, segui le istruzioni nella guida per l'utente di AWS Mainframe Modernization. L' BankDemo applicazione è un'applicazione mainframe in batch che crea e quindi avvia l'output di stampa.	Architetto del cloud

Configura un server di stampa LRS su Amazon EC2

Operazione	Description	Competenze richieste
Crea un'istanza Amazon EC2 Windows.	Per avviare un'istanza Amazon EC2 Windows, segui le istruzioni in Avvio di un' EC2	Architetto del cloud

Operazione	Description	Competenze richieste
	istanza Amazon nella EC2 documentazione di Amazon. Usa lo stesso nome host che hai usato per la licenza del prodotto LRS. L'istanza deve soddisfare i seguenti requisiti hardware e software per LRS VPSX/MFI: • CPU: dual core • RAM — 16 GB • Unità: 500 GB • EC2 Istanza minima: `m5.xlarge` • Sistema operativo: Windows • Software: Internet Informati on Services (IIS) o Apache **Note** I requisiti hardware e software precedent i sono destinati a un piccolo parco stampanti (circa 500-1000). Per ottenere i requisiti completi, rivolgiti ai tuoi contatti LRS e AWS.	

Operazione	Description	Competenze richieste
	1. Quando create l'istanza Windows, verificate che il EC2 nome host sia lo stesso usato per la licenza del prodotto LRS. 2. Connettiti alla tua EC2 istanza seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. 3. Nel menu Start di Windows, trova e apri Server Manager. 4. In Server Manager, scegli Dashboard, Quick Start, Aggiungi ruoli e funzionalità, quindi scegli Ruoli server. 5. In Ruoli server, scegli WebServer (IIS), quindi scegli Sviluppo applicazioni. 6. In Sviluppo di applicazioni, seleziona la casella di controllo CGI. 7. Per installare CGI, segui le istruzioni della procedura guidata per l'aggiunta di ruoli e funzionalità di Windows Server Manager. 8. Aprire la porta 5500 nel firewall di Windows dell'istanza per la EC2 comunicazione. LRS/Queue	

Operazione	Description	Competenze richieste
Installa LRS VPSX/MFI sull' EC2 istanza.	1. Connect alla tua EC2 istanza. 2. Apri il link alla pagina di download del prodotto contenuto nel messaggio e-mail di LRS che avresti dovuto ricevere. **Note** I prodotti LRS sono distribuiti tramite trasferim ento elettronico di file (EFT). 3. Scaricate LRS VPSX/MFI e decomprimete il file (cartella predefinita:). `c:\LRS` 4. Per installare LRS VPSX/ MFI, lanciate LRS Product Installer dalla cartella decompressa. 5. Nel menu Seleziona funzionalità, seleziona te VPSX® Server, quindi scegliete Avanti per avviare il processo di installaz ione. Una volta completata l'installazione, riceverete un messaggio di conferma.	Architetto del cloud

Operazione	Description	Competenze richieste
Installa LRS/Queue.	1. Connect alla tua EC2 istanza di Rocket Enterprise Server. 2. Aprite il link alla pagina di download del prodotto LRS dal messaggio e-mail di LRS che avreste dovuto ricevere, scaricate LRS/Queue, quindi decomprim ete il file. 3. Vai alla posizione in cui hai scaricato i file, quindi avvia LRS Product Installer per installare LRS/Queue. 4. Segui le istruzioni dell'LRS Product Installer per completare il processo di installazione.	Architetto del cloud

Operazione	Description	Competenze richieste
Installa LRS/DIS.	Il LRS/DIS prodotto è spesso incluso nell'installazione di LRS VPSX. Tuttavia, se non LRS/DIS è stato installat o insieme a LRS VPSX, seguite i seguenti passaggi per installarlo: 1. Connect alla tua VPSX/MFI EC2 istanza LRS. 2. Aprite il link alla pagina di download del prodotto LRS dal messaggio e-mail di LRS che avreste dovuto ricevere, scaricate LRS/DIS, quindi decomprimete il file. 3. Accedete alla posizione in cui avete scaricato i file, quindi avviate il programma di installazione del prodotto LRS. 4. In LRS Product Installer, espandete LRS Misc Tools, selezionate LRS DIS, quindi scegliete Avanti. 5. Seguite le altre istruzioni dell'LRS Product Installer per completare il processo di installazione.	Architetto del cloud

Operazione	Description	Competenze richieste
Crea un gruppo di destinazione.	Crea un gruppo target seguendo le istruzioni riportate in Creare un gruppo target per il tuo Network Load Balancer. Quando create il gruppo target, registrate l' VPSX/MFI EC2 istanza LRS come destinazione: 1. Nella pagina Specificare i dettagli del gruppo, per Scegli un tipo di destinazione, scegli Istanze. 2. Per Protocollo, scegli TCP. 3. Per Porta, scegli 5500. 4. Nella pagina Registra destinazioni, nella sezione Istanze disponibili, seleziona l'istanza LRS VPSX/MFI EC2 .	Architetto del cloud

Operazione	Description	Competenze richieste
Crea un Network Load Balancer.	Per creare il Network Load Balancer, segui le istruzioni nella documentazione di Elastic Load Balancing. Il Network Load Balancer indirizza il traffico da Rocket Enterprise Server all'istanza LRS. VPSX/MFI EC2 Quando create il Network Load Balancer, scegliete i seguenti valori nella pagina Listener and Routing: 1. Per Protocol (Protocollo), selezionare TCP. 2. Per Port, scegliete 5500. 3. Per Azione predefinita, scegli Inoltra a per il gruppo target che hai creato in precedenza.	Architetto del cloud

Integra Rocket Enterprise Server con LRS LRS/Queue VPSX/MFI

Operazione	Description	Competenze richieste
Configura Rocket Enterprise Server per LRS/Queue l'integrazione.	1. Connect alla tua EC2 istanza di Rocket Enterprise Server seguendo le istruzioni nella EC2 documentazione di Amazon. 2. Nel menu Start di Windows, apri l'interfaccia utente di	Architetto del cloud

Operazione	Description	Competenze richieste
	amministrazione di Rocket Enterprise Server. 3. Nella barra dei menu, scegli NATIVE. 4. Nel pannello di navigazione, scegli Directory Server, quindi scegli BANKDEMO per la tua regione Enterprise Server. 5. Da Generale, nel riquadro di navigazione a sinistra, scorri verso il basso fino alla sezione Aggiuntiva per configurare le variabili di ambiente (`LRSQ_ADDRESS` ,`LRSQ_PORT` ,`LRSQ_COMMAND`) in modo che puntino a LRSQ. • Per LRSQ_ADDRESS, inserisci l'indirizzo IP o il nome DNS del Network Load Balancer che hai creato in precedenza. • Per LRSQ_PORT, inserite VPSX LRSQ Listener Port (5500). • Per LRSQ_COMMAND, inserite la posizione del percorso dell'eseguibile LRSQ.	

Operazione	Description	Competenze richieste
	Note Attualmente LRS supporta un limite massimo di 50 caratteri per i nomi DNS. Se il nome DNS è più lungo di 50 caratteri, in alternativa è possibile utilizzar e l'indirizzo IP del Network Load Balancer.	

Operazione	Description	Competenze richieste
Configura Rocket Enterprise Server per l'integrazione con LRS. VPSX/MFI	1. Copia la `VPSX_MFI_R2` cartella dal programma di VPSX/MFI installazione di LRS nella posizione di Rocket Enterprise Server all'indirizzo. `C\BANKDEMO\print` 2. Connect alla tua EC2 istanza di Rocket Enterpris e Server seguendo le istruzioni nella EC2 documentazione di Amazon. 3. Nel menu Start di Windows, apri l'interfaccia utente di amministrazione di Rocket Enterprise Server. 4. Nella barra dei menu, scegli NATIVE. 5. Nel riquadro di navigazio ne, scegli Directory Server, quindi scegli BANKDEMO. 6. In BANKDEMO, scegli JES. 7. In JES Program Path, aggiungi il `DLL(VPSX_MFI_R2)` percorso da. `C\BANKDEMO\print`	Architetto del cloud

Configura la coda di stampa e gli utenti di stampa

Operazione	Description	Competenze richieste
Associate il modulo Rocket Software Print Exit al processo di esecuzione del server della stampante batch Rocket Enterprise Server.	1. Connect alla tua EC2 istanza di Rocket Enterprise Server seguendo le istruzioni nella EC2 documentazione di Amazon. 2. Nel menu Start di Windows, apri l'interfaccia utente di amministrazione di Rocket Focus Enterprise Server. 3. Nella barra dei menu, scegli NATIVE. 4. Nel riquadro di navigazione, scegli Directory Server, quindi scegli BANKDEMO. 5. In BANKDEMO, scegli JES e scorri verso il basso fino a Stampanti. 6. Nelle stampanti, associa il modulo Rocket Software Print Exit (per Batch) alla stampante LRSPRTE6 batch Rocket Enterprise Server Server Server Execution Process (SEP). Ciò consente il routing dell'output di stampa verso LRS VPSX/MFI. Per ulteriori informazioni sulla configurazione, vedere Using the Exit nella	Architetto del cloud

Operazione	Description	Competenze richieste
	documentazione di Micro Focus. OpenText	
Crea una coda di output di stampa in LRS VPSX/MFI e integrala con LRS X. PageCenter	1. Connect alla tua VPSX/MFI EC2 istanza LRS. 2. Nel menu Start di Windows, aprite l'interfaccia Web VPSX. 3. Nel riquadro di navigazione, scegli Stampanti. 4. Scegli Aggiungi, quindi scegli Aggiungi stampante. 5. Nella pagina di configura zione della stampante , in Nome stampante, immettete Locale. 6. Per ID VPSX, immettere. VPS1 7. Per CommType, seleziona te TCPIP/LRSQ. 8. Per l'indirizzo host/IP, inserite l'indirizzo IP del Network Load Balancer che fronteggia le istanze LRS X. PageCenter EC2 9. Per Porta remota, immettere 5800. 10. Per Remote queue, inserite il nome della cartella dei documenti LRS PageCente r X in cui verrà memorizzato l'output. 11. Scegliere Aggiungi.	Architetto del cloud

Operazione	Description	Competenze richieste
Crea un utente di stampa in LRS VPSX/MFI.	1. Connect alla tua VPSX/MFI EC2 istanza LRS. 2. Nel menu Start di Windows, aprite l'interfaccia Web VPSX. 3. Nel riquadro di navigazio ne, scegli Sicurezza, quindi scegli Utenti. 4. Nella colonna Nome utente, scegli admin, quindi scegli Copia. 5. Nella finestra Gestione del profilo utente, per Nome utente, inserisci un nome utente (ad esempio, PrintUser). 6. Per Descrizione, inserisci una breve descrizione (ad esempio, Utente per la stampa di prova). 7. Scegliere Aggiorna. In questo modo viene creato un utente di stampa (ad esempio, PrintUser). 8. Nel riquadro di navigazio ne, in Utente, scegli il nuovo utente che hai creato. 9. Nel menu Comando, scegli Sicurezza. 10. Nella pagina Regole di sicurezza, scegli tutte le opzioni di sicurezza della stampante e del lavoro	Architetto del cloud

Operazione	Description	Competenze richieste
	applicabili, quindi scegli Salva. 11. Per aggiungere il nuovo utente di stampa al gruppo Amministratore, nel pannello di navigazione, scegli Sicurezza, quindi scegli Configura. 12. Nella finestra di configura zione della sicurezza, aggiungi il tuo nuovo utente di stampa alla colonna Amministratore.	

Configura un server LRS PageCenter X su Amazon EC2

Operazione	Description	Competenze richieste
Crea un'istanza Amazon EC2 Windows.	Avvia un'istanza Amazon EC2 Windows seguendo le istruzioni del Passaggio 1: Avvia un'istanza nella EC2 documentazione di Amazon. Usa lo stesso nome host che hai usato per la licenza del prodotto LRS. L'istanza deve soddisfare i seguenti requisiti hardware e software per PageCenter LRS X: • CPU: Dual Core • RAM — 16 GB	Architetto del cloud

Operazione	Description	Competenze richieste
	• Unità: 500 GB • EC2 Istanza minima: `m5.xlarge` • Sistema operativo: Windows • Software: IIS o Apache **Note** I requisiti hardware e software precedent i sono destinati a un piccolo parco stampanti (circa 500-1000). Per ottenere i requisiti completi, rivolgiti ai tuoi contatti LRS e AWS. 1. Quando create l'istanza Windows, verificate che il EC2 nome host sia lo stesso usato per la licenza del prodotto LRS. 2. Connect alla tua EC2 istanza seguendo le istruzioni nella EC2 documentazione di Amazon. 3. Nel menu Start di Windows, trova e apri Server Manager.	

Operazione	Description	Competenze richieste
	4. In Server Manager, scegli Dashboard, Quick Start, Aggiungi ruoli e funzionalità, quindi scegli Ruoli server. 5. In Ruoli server, scegli WebServer (IIS), quindi scegli Sviluppo applicazioni. 6. In Sviluppo di applicazioni, seleziona la casella di controllo CGI. 7. Per installare CGI, segui le istruzioni della procedura guidata per l'aggiunta di ruoli e funzionalità di Windows Server Manager. 8. Apri la porta 5800 per il TCP/IP traffico in entrata nel firewall Windows dell'istanza. EC2 LRS VPSX utilizza il TCPIP/LRS Q protocollo sulla porta 5800 per comunicare con LRS X. PageCenter	

Operazione	Description	Competenze richieste
Installa LRS PageCenter X sull' EC2 istanza.	1. Connect alla tua EC2 istanza. 2. Apri il link alla pagina di download del prodotto contenuto nel messaggio e-mail di LRS che avresti dovuto ricevere. **Note** I prodotti LRS sono distribuiti tramite trasferim ento elettronico di file (EFT). 3. Scaricate LRS PageCente r X e decomprimete il file (cartella predefinita:). `c:\LRS` 4. Per installare LRS PageCenter X, lanciate LRS Product Installer dalla cartella decompressa. 5. Nel menu Seleziona funzionalità, seleziona te PageCenterX, quindi scegliete Avanti per avviare il processo di installaz ione. Una volta completata l'installazione, riceverai un messaggio di conferma.	Architetto del cloud

Operazione	Description	Competenze richieste
Installa LRS/DIS.	Il LRS/DIS prodotto è spesso incluso nell'installazione di LRS VPSX. Tuttavia, se non LRS/DIS è stato installat o insieme a LRS VPSX, seguite i seguenti passaggi per installarlo: 1. Connect alla tua EC2 istanza LRS PageCenter X. 2. Aprite il link alla pagina di download del prodotto LRS dall'e-mail LRS che avreste dovuto ricevere, scaricate LRS/DIS, quindi decomprim ete il file. 3. Accedete alla posizione in cui avete scaricato i file, quindi avviate il programma di installazione del prodotto LRS. 4. In LRS Product Installer, espandete LRS Misc Tools, selezionate LRS DIS, quindi scegliete Avanti. 5. Seguite le altre istruzioni dell'LRS Product Installer per completare il processo di installazione.	Architetto del cloud

Operazione	Description	Competenze richieste
Crea un gruppo di destinazione.	Crea un gruppo target seguendo le istruzioni riportate in Creare un gruppo target per il tuo Network Load Balancer. Quando create il gruppo target, registrate l' EC2 istanza LRS PageCenter X come destinazione: 1. Nella pagina Specificare i dettagli del gruppo, per Scegli un tipo di destinazione, scegliete Istanze. 2. Per Protocollo, scegli TCP. 3. Per Porta, scegli 5800. 4. Nella pagina Registra destinazioni, nella sezione Istanze disponibili, selezionate l'istanza LRS PageCenter X. EC2	Architetto del cloud

Operazione	Description	Competenze richieste
Crea un Network Load Balancer.	Per creare il Network Load Balancer, segui le istruzioni nella documentazione di Elastic Load Balancing. Il Network Load Balancer indirizza il traffico da LRS VPSX/MFI all'istanza LRS X. PageCenter EC2 Quando create il Network Load Balancer, scegliete i seguenti valori nella pagina Listener and Routing: 1. Per Protocol (Protocollo), selezionare TCP. 2. Per Port, scegliete 5800. 3. Per Azione predefinita, scegli Inoltra a per il gruppo target che hai creato in precedenza.	Architetto del cloud

Configura le funzionalità di gestione dell'output in LRS X PageCenter

Operazione	Description	Competenze richieste
Abilita la funzione di importazione in LRS X. PageCenter	È possibile utilizzare la funzione LRS PageCenter X Import per riconoscere gli output che arrivano su LRS PageCenter X in base a criteri come Job name o Form ID. Potete quindi indirizzare gli	Architetto del cloud

Operazione	Description	Competenze richieste
	output verso cartelle specifiche in LRS X. PageCenter Per abilitare la funzione di importazione, effettuate le seguenti operazioni: 1. Connect alla tua EC2 istanza LRS PageCenter X seguendo le istruzioni nella EC2 documentazione di Amazon. 2. Nel menu Start di Windows, apri l'interfaccia Web PCX. 3. In Folder Explorer, scegliete Amministratore. 4. Nella pagina Configurazione, scegli Avanzate, Parametro di importazione. 5. Nella sezione Parametri di importazione, seleziona la casella di controllo Importazione avanzata. 6. Per confermare le modifiche, scegliete Aggiorna.	

Operazione	Description	Competenze richieste
Configura la politica di conservazione dei documenti.	LRS PageCenter X utilizza una politica di conservazione dei documenti per decidere per quanto tempo conservare un documento in PageCenter LRS X. Per configurare la politica di conservazione dei documenti , effettuate le seguenti operazioni: 1. Connect alla tua EC2 istanza LRS PageCenter X. 2. Nel menu Start di Windows, aprite PCX Web Interface. 3. In Folder Explorer, scegliete Amministratore. 4. Nella pagina di amministr azione, scegli Archive Group List/General admin, quindi scegli Retention policy. 5. Nella sezione Politica di conservazione, scegli Aggiungi per creare una politica di conservazione. 6. Nella pagina Informazioni sulla politica di conservaz ione, inserisci il nome, la descrizione e il periodo di conservazione del documento della politica di conservazione.	Architetto del cloud

Operazione	Description	Competenze richieste
	7. Per salvare le modifiche e creare la politica, scegli Ok.	

Operazione	Description	Competenze richieste
Create una regola per indirizzare il documento di output verso una cartella specifica in LRS PageCenter X.	In LRS PageCenter X, Destination determina il percorso della cartella in cui verrà inviato l'output quando questa destinazione viene richiamata da Report Definition. Per questo esempio, create una cartella basata sulla cartella Form ID nella definizione del report e salvate l'output in quella cartella. 1. Connect alla tua EC2 istanza LRS PageCenter X. 2. Nel menu Start di Windows, aprite PCX Web Interface. 3. In Folder Explorer, scegliete Amministratore, Importazione avanzata, Destinazione. 4. Nella sezione Destinazione, scegli Aggiungi per aprire il modulo Manutenzione della destinazione. 5. Nel modulo Manutenzione della destinazione, inserisci i seguenti valori: • Nome della destinazione: modulo • Descrizione: descrizione della destinazione, ad esempio la struttura delle cartelle basata su moduli	Architetto del cloud

Operazione	Description	Competenze richieste
	• Tipo di destinazione: cartella • Parametri della cartella: percorso della cartella di importazione (il percorso della cartella che verrà creato in PageCenter X all'arrivo del documento; ad esempio, il percorso **`/Test/&FORM/&IMPORTDATE/&IMPORTTIME`** creerà una **`Test`** cartella base, una sottocartella basata sul nome Form-Id **`STD`**, una sottocartella basata sulla data di importazione e quindi una sottocartella basata sull'ora di importazione) • Nome del documento : nome dinamico assegnato a un documento quando è archiviato nella cartella. 6. Nell'elenco a discesa, scegli una politica di conservazione. Ad esempio, scegli Anno1 per conservare il documento per 1 anno. 7. Per salvare le modifiche, scegliete Ok.	

Operazione	Description	Competenze richieste
Crea una definizione di report.	1. Connect alla tua EC2 istanza LRS PageCenter X. 2. Nel menu Start di Windows, aprite PCX Web Interface. 3. In Folder Explorer, scegliete Admin, Advance Import, Report Definition, quindi scegliete Aggiungi. 4. Nella pagina Gestione della definizione del report, nella scheda Generale, inserisci il nome della definizione del report. 5. Nella scheda Generale, in Campi, è possibile specifica re criteri di selezione come Job Name, Form, Class e Author. Ad esempio, è possibile inserire un Job Name di MFIDEMO. Il valore Job Name sarà il nome del processo batch che genererà l'output di stampa. 6. Nella scheda Destinazione, in Destinazione disponibi le, scegliete la destinazione precedentemente creata (Modulo). 7. Scegli Aggiungi per aggiungere la destinazione del modulo come destinazi one assegnata.	Architetto cloud

Operazione	Description	Competenze richieste
	Note Questo esempio include una definizione di report in cui un output generato da MFIDEMO e indirizzato a LRS PageCente r X viene salvato nella struttura di cartelle definita nella definizione di destinazione.	

Configura l'autenticazione e l'autorizzazione per la gestione dell'output

Operazione	Description	Competenze richieste
Crea un dominio AWS Managed Microsoft AD con utenti e gruppi.	1. Per creare una directory su AWS Managed Microsoft AD, segui le istruzioni in Crea la tua directory AWS Managed Microsoft AD. 2. Per distribuire un' EC2 istanza (Active Directory manager) e installare gli strumenti di Active Directory per gestire AWS Managed Microsoft AD, segui le istruzioni nella Fase 3: Distribuisci un' EC2 istanza	Architetto del cloud

Operazione	Description	Competenze richieste
	per gestire AWS Managed Microsoft AD. 3. Per connetterti alla tua EC2 istanza, segui le istruzioni nella EC2 documentazione di Amazon. **Note** Quando ti connetti all' EC2 istanza, nella finestra Sicurezza di Windows, inserisci le credenziali di amministratore per la directory che hai creato nel passaggio 1. 4. Nel menu Start di Windows, in Strumenti di amministrazione di Windows, scegli Utenti e computer di Active Directory. 5. Per creare un utente di stampa nel dominio Active Directory, segui le istruzioni in Creare un utente.	

Operazione	Description	Competenze richieste
Unisci le EC2 istanze a un dominio AWS Managed Microsoft AD.	Unisci le EC2 istanze LRS VPSX/MFI e LRS PageCenter X al tuo dominio AWS Managed Microsoft AD automaticamente (documentazione AWS Knowledge Center) o manualmente (documentazione AWS Directory Service).	Architetto del cloud
Configura e integra LRS/DIS con AWS Managed Microsoft AD per l' EC2 istanza LRS PageCenter X.	1. Connect alla tua EC2 istanza LRS PageCenter X. 2. Nel menu Start di Windows, aprite PCX Web Interface. 3. In Folder Explorer, scegliete Amministratore. 4. Nella pagina Configurazione, nella sezione Parametri di sicurezza, per Tipo di sicurezza, seleziona LRS/DIS. 5. Immettete le vostre preferenze per le altre opzioni nella sezione Parametri di sicurezza. 6. Nel menu Start di Windows, apri la cartella PageCenterX, scegli Server Start, quindi scegli Server Stop. 7. Accedete a LRS PageCenter X con il nome utente e la password di Active Directory.	Architetto del cloud

Operazione	Description	Competenze richieste
Configura un gruppo di importazione per importare l'output da LRS VPSX a LRS X. PageCenter	1. Connect alla tua EC2 istanza LRS PageCenter X. 2. Nel menu Start di Windows, aprite PCX Web Interface. 3. In Folder Explorer, scegliete Admin, Security admin, Groups. 4. Nella sezione Gruppi, scegli Aggiungi per aprire il modulo delle preferenze di gruppo. 5. Nel modulo di preferenza di gruppo, inserisci i valori per il nome del gruppo e la descrizione. 6. Espandi le opzioni generali, quindi seleziona la casella di controllo Importa. 7. Per salvare le modifiche, scegliete Ok.	Architetto del cloud
Aggiungi una regola di sicurezza al gruppo Import.	1. Aprire il menu contestua le (fare clic con il pulsante destro del mouse) per il gruppo Importa. 2. Scegli Avanzate, quindi scegli Sicurezza. 3. Nella sezione Sicurezza, scegli Importa e seleziona la casella di controllo Sottocartella. 4. Per salvare le modifiche, scegli Applica.	Architetto del cloud

Operazione	Description	Competenze richieste
Crea un utente in LRS PageCenter X per eseguire l'importazione dell'output da LRS VPSX/MFI.	Quando create un utente in LRS PageCenter X per eseguire l'importazione dell'output, il nome utente deve essere lo stesso dell'ID VPSX della coda di output di stampa in LRS VPSX/MFI. In questo esempio, l'ID VPSX è. VPS1 1. Connect alla tua EC2 istanza LRS PageCenter X. 2. Nel menu Start di Windows, aprite PCX Web Interface. 3. In Folder Explorer, scegliete Admin, Security admin, User. 4. Scegli Aggiungi per aprire il modulo di manutenzione del profilo utente. 5. In Manutenzione del profilo utente, per Nome utente, inserisci VPS1.	Architetto del cloud

Operazione	Description	Competenze richieste
Aggiungi l'utente LRS PageCenter X Import al gruppo di sola importazione.	Per fornire le autorizzazioni necessarie per l'importazione di documenti da LRS VPSX a LRS X, effettuate le seguenti operazioni PageCenter: 1. Connect alla tua EC2 istanza LRS PageCenter X. 2. Nel menu Start di Windows, aprite PCX Web Interface. 3. In Folder Explorer, scegliete Admin, Security admin, Groups. 4. Nella sezione Gruppi, apri il menu contestuale (fai clic con il pulsante destro del mouse) per il gruppo Importa solo importazione, quindi scegli Avanzata, Sicurezza. 5. Nella pagina Folder Security Records (ImportOnly), scegli la scheda Utente. 6. Nella scheda Utente, in Nome, seleziona utente VPS1dall'elenco a discesa e scegli Applica.	Architetto del cloud

Operazione	Description	Competenze richieste
Configura LRS/DIS con AWS Managed Microsoft AD per l' VPSX/MFI EC2 istanza LRS.	1. Connect alla tua VPSX/MFI EC2 istanza LRS. 2. Nel menu Start di Windows, aprite l'interfaccia Web VPSX. 3. Nel riquadro di navigazio ne, scegli Sicurezza, quindi scegli Configura. 4. Nella pagina Configura zione della sicurezza, nella sezione Parametri di sicurezza, per Tipo di sicurezza, seleziona LRS/ DIS (esterno). 5. Immettete le vostre preferenze per le altre opzioni nella sezione Parametri di sicurezza. 6. Nel menu Start di Windows, aprite la cartella LRS Output Management, scegliete Server Start, quindi scegliete Server Stop. 7. Accedete a LRS VPSX/ MFI con il nome utente e la password di Active Directory.	Architetto del cloud

Configura Amazon FSx for Windows File Server come archivio dati operativo per LRS X PageCenter

Operazione	Description	Competenze richieste
Create un file system per LRS X. PageCenter	Per utilizzare Amazon FSx for Windows File Server come archivio dati operativo per LRS PageCenter X in un ambiente Multi-AZ, segui le istruzioni nel Passaggio 1: Crea il tuo file system.	Architetto del cloud
Mappa la condivisione di file sull' EC2 istanza LRS PageCenter X.	Per mappare la condivisione di file creata nel passaggio precedente all' EC2 istanza LRS PageCenter X, seguite le istruzioni nel Passo 2: Mappate la condivisione di file su un' EC2 istanza che esegue Windows Server.	Architetto del cloud
Mappa LRS PageCenter X Control Directory e Master Folder Directory sull'unità condivisa FSx di rete Amazon.	1. Connect alla tua EC2 istanza LRS PageCente r X seguendo le istruzioni nella EC2 documentazione di Amazon. 2. Nel menu Start di Windows, apri l'interfaccia Web PCX. 3. In Folder Explorer, scegliete Amministratore, Configura zione. 4. Nella pagina Configura zione, scegli Directory , quindi scegli Control Directory.	Architetto del cloud

Operazione	Description	Competenze richieste
	5. In Control Directories, immettete. `\\FSx file share DNS name\share\cntl` 6. In Master Folder Directory, immettete`\\FSx file share DNS name\share\mstr` .	

Prova un flusso di lavoro per la gestione dell'output

Operazione	Description	Competenze richieste
Avvia una richiesta di stampa in batch dall'app Rocket Software BankDemo .	1. Apri l'emulatore di terminale 3270 nell'istanza di Rocket Enterprise Server. EC2 2. Connect all' BankDemo app eseguendo il comando`connect 127.0.0.1:9278` . 3. Nell'interfaccia della BankDemo riga di comando, per ID utente, inserisci B0001. Per Password, immettere una chiave non vuota. 4. Per l'opzione Richiedi dichiarazioni stampate, immettete X nella riga vuota. 5. Nella sezione Invia dichiarazione per, per Mail,	Tecnico collaudatore

Operazione	Description	Competenze richieste
	inserisci Y, quindi premi F10.	

Operazione	Description	Competenze richieste
Controllate l'output di stampa in LRS X. PageCenter	1. Connect alla tua EC2 istanza LRS PageCente r X seguendo le istruzioni nella EC2 documentazione di Amazon. 2. Nel menu Start di Windows, apri l'interfaccia Web PCX. 3. Nel pannello di navigazione, aprite la cartella Test, aprite la cartella STD, quindi aprite la cartella con la data di esecuzione del processo, ad esempio `08-03-2023` (MM-GG-AAAA). **Note** Questa è la stessa struttura di cartelle definita nella storia Create a rule to routing del documento di output verso una cartella specifica in LRS X. PageCenter 4. Apri il file `formtest-STD.txt` . Ora potete vedere l'output di stampa di un estratto conto con le colonne per il numero di conto. , Descrizio ne, data, importo e saldo.	Tecnico collaudatore

Operazione	Description	Competenze richieste
	Per un esempio, vedi l'`batch_print_output` allegato di questo modello.	

Risorse correlate

- LRS
- Flusso di dati di presentazione delle funzioni avanzate (documentazione IBM)
- Line Conditioned Data Stream (LCDS) (documentazione Compart)
- Potenziamento dei carichi di lavoro mainframe aziendali su AWS con Micro Focus (post sul blog)
- Modernizza i tuoi carichi di lavoro di stampa online mainframe su AWS (AWS Prescriptive Guidance)
- Modernizza i tuoi carichi di lavoro di stampa in batch mainframe su AWS (AWS Prescriptive Guidance)

Informazioni aggiuntive

Considerazioni

Durante il tuo percorso di modernizzazione, potresti prendere in considerazione un'ampia varietà di configurazioni per i processi mainframe in batch e online e l'output che generano. La piattaforma mainframe è stata personalizzata da ogni cliente e fornitore che la utilizza con requisiti particolari che influiscono direttamente sulla stampa. Ad esempio, la piattaforma attuale potrebbe incorporare il flusso di dati IBM AFP o gli LCD Xerox nel flusso di lavoro corrente. Inoltre, i caratteri di controllo del mainframe carriage e le parole dei comandi del canale possono influire sull'aspetto della pagina stampata e potrebbero richiedere una gestione speciale. Come parte del processo di pianificazione della modernizzazione, consigliamo di valutare e comprendere le configurazioni del proprio ambiente di stampa specifico.

Acquisizione dei dati di stampa

Rocket Software Print Exit trasmette le informazioni necessarie a LRS per VPSX/MFI elaborare efficacemente il file di spool. Le informazioni sono costituite da campi passati nei blocchi di controllo pertinenti, come i seguenti:

- NOME DEL LAVORO
- PROPRIETARIO (USERID)
- DESTINAZIONE
- MODULO
- NOME DEL FILE
- SCRITTORE

LRS VPSX/MFI supporta i seguenti meccanismi batch mainframe per l'acquisizione di dati da Rocket Enterprise Server:

- Elaborazione COBOL in BATCH utilizzando istruzioni JCL print/spool SYSOUT standard. z/OS DD/OUTPUT
- Elaborazione COBOL print/spool in BATCH utilizzando istruzioni z/OS JCL CA-SPOOL SUBSYS DD standard.
- IMS/COBOL print/spoolelaborazione mediante l'interfaccia CBLTDLI. Per un elenco completo dei metodi supportati e degli esempi di programmazione, consultate la documentazione di LRS inclusa nella licenza del prodotto.

Controlli dello stato del parco stampanti

LRS VPSX/MFI (LRS LoadX) è in grado di eseguire controlli approfonditi dello stato di salute, tra cui la gestione dei dispositivi e l'ottimizzazione operativa. La gestione dei dispositivi è in grado di rilevare guasti in un dispositivo di stampa e indirizzare la richiesta di stampa a una stampante funzionante. Per ulteriori informazioni sui controlli approfonditi dello stato delle flotte di stampanti, consultate la documentazione LRS inclusa nella licenza del prodotto.

Autenticazione e autorizzazione alla stampa

LRS/DIS enables LRS applications to authenticate user IDs and passwords by using Microsoft Active Directory or a Lightweight Directory Access Protocol (LDAP) server. In addition to basic print authorization, LRS/DISpuò anche applicare controlli di sicurezza di stampa a livello granulare nei seguenti casi d'uso:

- Gestisci chi può sfogliare il lavoro della stampante.
- Gestisci il livello di navigazione dei lavori di altri utenti.

- Gestisci le attività operative, ad esempio la sicurezza a livello di comando come il blocco o il rilascio, l'eliminazione, la modifica, la copia e il reindirizzamento. La sicurezza può essere configurata dall'ID utente o dal gruppo, in modo simile a un gruppo di sicurezza di Active Directory o a un gruppo LDAP.

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Modernizza i carichi di lavoro di stampa in batch su mainframe AWS utilizzando Rocket Enterprise Server e LRS VPSX/MFI

Shubham Roy e Kevin Yung, Amazon Web Services

Abraham Rondon, Micro Focus

Guy Tucker, Levi, Ray and Shoup Inc

Riepilogo

Questo modello mostra come modernizzare i carichi di lavoro di stampa in batch mainframe critici per l'azienda sul cloud Amazon Web Services (AWS) utilizzando Rocket Enterprise Server come runtime per un'applicazione mainframe modernizzata e LRS VPSX/MFI (Micro Focus Interface) come server di stampa. Il modello si basa sull'approccio di modernizzazione del mainframe replatform. In questo approccio, esegui la migrazione dei lavori in batch del mainframe su Amazon Elastic Compute Cloud (Amazon EC2) e la migrazione del database mainframe, ad esempio IBM for DB2) z/OS, to Amazon Relational Database Service (Amazon RDS). The authentication and authorization for the modernized print workflow is performed by AWS Directory Service for Microsoft Active Directory, also known as AWS Managed Microsoft AD. The LRS Directory Information Server (LRS/DIS è integrato con AWS Managed Microsoft AD. Modernizzando i carichi di lavoro di stampa in batch, puoi ridurre i costi dell'infrastruttura IT, mitigare il debito tecnico legato alla manutenzione dei sistemi legacy, rimuovere i silos di dati, aumentare l'agilità e l'efficienza con un DevOps modello e sfruttare le risorse e l'automazione su richiesta nel cloud AWS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un carico di lavoro di stampa o gestione dell'output su mainframe
- Conoscenze di base su come ricostruire e fornire un'applicazione mainframe eseguibile su Rocket Enterprise Server (per ulteriori informazioni, consultate la scheda tecnica di Rocket Enterprise Server nella documentazione di Rocket).
- Conoscenza di base delle soluzioni e dei concetti di stampa su cloud di LRS
- Software e licenza Rocket Enterprise Server (per ulteriori informazioni, contattate il reparto vendite di Rocket).

- VPSX/MFI, LRS/Queue, and LRS/DISSoftware e licenze LRS (per ulteriori informazioni, contattate il reparto vendite LRS).

> **Note**
>
> Per ulteriori informazioni sulle considerazioni sulla configurazione per i carichi di lavoro di stampa in batch su mainframe, vedere Considerazioni nella sezione Informazioni aggiuntive di questo modello.

Versioni del prodotto

- Rocket Enterprise Server 6.0 (aggiornamento del prodotto 7)
- LRS VPSX/MFI V1R3 o versione successiva

Architecture

Stack tecnologico di origine

- Sistema operativo: IBM z/OS
- Linguaggio di programmazione: Common Business-Oriented Language (COBOL), Job Control Language (JCL) e Customer Information Control System (CICS)
- Database: IBM DB2 for and Virtual Storage Access Method (z/OS VSAM)
- Sicurezza: Resource Access Control Facility (RACF), CA Top Secret for z/OS e Access Control Facility 2 () ACF2
- Gestione della stampa e dell'output: prodotti di z/OS stampa mainframe IBM (IBM Tivoli Output Manager for z/OS, LRS e CA View)

Stack tecnologico Target

- Sistema operativo: Microsoft Windows Server in esecuzione su Amazon EC2
- Elaborazione — Amazon EC2
- Linguaggio di programmazione: COBOL, JCL e CICS
- Banca dati — Amazon RDS
- Sicurezza: AWS Managed Microsoft AD

- Gestione della stampa e dell'output: soluzione di stampa LRS su AWS
- Ambiente di runtime mainframe: Rocket Enterprise Server

Architettura di origine

Il diagramma seguente mostra una tipica architettura allo stato attuale per un carico di lavoro di stampa in batch su mainframe:

Il diagramma mostra il flusso di lavoro seguente:

1. Gli utenti eseguono transazioni commerciali su un sistema di coinvolgimento (SoE) basato su un'applicazione IBM CICS scritta in COBOL.
2. Il SoE richiama il servizio mainframe, che registra i dati delle transazioni commerciali in un database system-of-records (SoR) come IBM for z/OS. DB2
3. Il SoR conserva i dati aziendali del SoE.
4. Il batch job scheduler avvia un processo batch per generare l'output di stampa.
5. Il processo batch estrae i dati dal database, li formatta in base ai requisiti aziendali e quindi genera risultati aziendali come estratti conto di fatturazione, carte d'identità o estratti conto di prestito. Infine, il processo in batch indirizza l'output alla gestione dell'output di stampa per l'elaborazione e la consegna dell'output, in base ai requisiti aziendali.
6. La gestione dell'output di stampa riceve l'output di stampa dal processo batch e quindi lo consegna a una destinazione specifica, ad esempio e-mail, una condivisione di file che utilizza FTP sicuro, una stampante fisica che utilizza soluzioni di stampa LRS (come illustrato in questo modello) o IBM Tivoli.

Architettura Target

Il diagramma seguente mostra un'architettura per un carico di lavoro di stampa in batch mainframe distribuito nel cloud AWS:

Il diagramma mostra il flusso di lavoro seguente:

1. Il batch job scheduler avvia un processo batch per creare output di stampa, come estratti conto di fatturazione, carte d'identità o rendiconti di prestito.

2. Il processo batch del mainframe (riplatformato su Amazon EC2) utilizza il runtime di Rocket Enterprise Server per estrarre i dati dal database dell'applicazione, applicare la logica di business ai dati, formattare i dati e quindi inviarli a una destinazione di stampa utilizzando Rocket Software Print Exit (documentazione Micro Focus).
3. Il database dell'applicazione (un SoR eseguito su Amazon RDS) mantiene i dati per l'output di stampa.
4. La soluzione di VPSX/MFI stampa LRS è implementata su Amazon EC2 e i suoi dati operativi sono archiviati in Amazon Elastic Block Store (Amazon EBS). LRS VPSX/MFI utilizza l'agente di TCP/IP-based LRS/Queue trasmissione per raccogliere dati di stampa tramite l'API JES Print Exit di Rocket Software e consegnarli a una destinazione di stampa specificata.

> **Note**
>
> La soluzione di destinazione in genere non richiede modifiche alle applicazioni per adattarsi ai linguaggi di formattazione del mainframe, come IBM Advanced Function Presentation (AFP) o Xerox Line Condition Data Stream (LCDS). Per ulteriori informazioni sull'utilizzo di Rocket Software per la migrazione e la modernizzazione delle applicazioni mainframe su AWS, consulta il post di blog Empowering Enterprise Mainframe Workloads on AWS with Micro Focus.

Architettura dell'infrastruttura AWS

Il diagramma seguente mostra un'architettura di infrastruttura AWS altamente disponibile e sicura per un carico di lavoro di stampa in batch mainframe:

Il diagramma mostra il flusso di lavoro seguente:

1. Lo scheduler batch avvia il processo batch e viene distribuito su Amazon EC2 su più zone di disponibilità per l'alta disponibilità (HA).

> **Note**
>
> Questo modello non copre l'implementazione dello scheduler in batch. Per ulteriori informazioni sull'implementazione, consultate la documentazione del fornitore del software relativa allo scheduler in uso.

2. Il processo batch del mainframe (scritto in un linguaggio di programmazione come JCL o COBOL) utilizza la logica aziendale principale per elaborare e generare output di stampa, come estratti conto di fatturazione, carte d'identità e dichiarazioni di prestito. Il processo viene distribuito su Amazon EC2 in due zone di disponibilità per HA e utilizza Rocket Software Print Exit per indirizzare l'output di stampa a LRS VPSX/MFI per la stampa dell'utente finale.
3. LRS utilizza un comando. VPSX/MFI TCP/IP-based LRS/Queue transmission agent to collect or capture print data from the Rocket Software JES Print Exit programming interface. Print Exit passes the necessary information to enable LRS VPSX/MFI to effectively process the spool file and dynamically build LRS/Queue I comandi vengono quindi eseguiti utilizzando una funzione standard integrata di Rocket Software.

> **Note**
>
> Per ulteriori informazioni sui dati di stampa trasferiti da Rocket Software Print Exit a LRS/Queue e sui meccanismi batch mainframe VPSX/MFI supportati da LRS, consultate Print Data Capture nella sezione Informazioni aggiuntive di questo modello.

4.
> **Note**
>
> Un Network Load Balancer fornisce un nome DNS per integrare Rocket Enterprise Server con LRS VPSX/MFI. : LRS VPSX/MFI supports a Layer 4 load balancer. The Network Load Balancer also does a basic health check on LRS VPSX/MFI e indirizza il traffico verso i target registrati che sono integri.

5.
> **Note**
>
> Il server di VPSX/MFI stampa LRS è distribuito EC2 su Amazon in due zone di disponibilità per HA e utilizza Amazon EBS come archivio dati operativo. LRS VPSX/MFI supporta le modalità di servizio attivo-attivo e attivo-passivo. Questa architettura utilizza più elementi AZs in una coppia attivo-passiva come standby attivo e hot standby. Il Network Load Balancer esegue un controllo dello stato VPSX/MFI EC2 delle istanze LRS e indirizza il traffico verso le istanze hot standby nell'altra AZ se un'istanza attiva non è integra. Le richieste di stampa vengono mantenute nella LRS Job Queue localmente in ciascuna istanza. EC2 In caso di ripristino, è necessario riavviare un'istanza fallita affinché i servizi LRS riprendano l'elaborazione della richiesta di stampa. : LRS VPSX/MFI può anche eseguire controlli di integrità a livello di parco stampanti. Per ulteriori informazioni, vedete

Controlli dello stato del parco stampanti nella sezione Informazioni aggiuntive di questo modello.

6. AWS Managed Microsoft AD si integra con LRS/DIS per eseguire l'autenticazione e l'autorizzazione del flusso di lavoro di stampa. Per ulteriori informazioni, consulta Autenticazione e autorizzazione alla stampa nella sezione Informazioni aggiuntive di questo modello.
7. LRS VPSX/MFI utilizza Amazon EBS per lo storage a blocchi. Puoi eseguire il backup dei dati di Amazon EBS da EC2 istanze attive su Amazon S3 point-in-time come snapshot e ripristinarli su volumi EBS in hot standby. Per automatizzare la creazione, la conservazione e l'eliminazione degli snapshot di volume Amazon EBS, puoi utilizzare Amazon Data Lifecycle Manager per impostare la frequenza degli snapshot automatici e ripristinarli in base ai requisiti RTO/RPO.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Block Store (Amazon EBS) fornisce volumi di storage a livello di blocco da utilizzare con le istanze. EC2 Il comportamento dei volumi EBS è simile a quello dei dispositivi a blocchi non formattati e non elaborati. Puoi montare questi volumi come dispositivi sulle istanze.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi usare Amazon EC2 per lanciare tutti o pochi server virtuali di cui hai bisogno, con scalabilità orizzontale o orizzontale.
- Amazon Relational Database Service (Amazon RDS) è un servizio Web che semplifica la configurazione, il funzionamento e la scalabilità di un database relazionale nel cloud AWS. Fornisce una capacità ridimensionabile e conveniente per un database relazionale e gestisce le attività di amministrazione comuni del database.
- AWS Directory Service per Microsoft Active Directory, noto anche come AWS Managed Microsoft AD, consente ai carichi di lavoro basati sulle directory e alle risorse AWS di utilizzare Microsoft Active Directory nel cloud AWS.

Altri strumenti

- LRS VPSX/MFI (Micro Focus Interface), sviluppato congiuntamente da LRS e Rocket Software, acquisisce l'output da una bobina JES di Rocket Enterprise Server e lo consegna in modo affidabile a una destinazione di stampa specificata.

- LRS Directory Information Server (LRS/DIS) viene utilizzato per l'autenticazione e l'autorizzazione durante il flusso di lavoro di stampa.
- TCP/IP-based LRS/Queue transmission agent is used by LRS VPSX/MFIper raccogliere o acquisire dati di stampa tramite l'interfaccia di programmazione JES Print Exit di Rocket Software.
- Rocket Enterprise Server è un ambiente di distribuzione delle applicazioni per applicazioni mainframe. Fornisce l'ambiente di esecuzione per le applicazioni mainframe che vengono migrate o create utilizzando qualsiasi versione di Rocket Software Enterprise Developer.

Epiche

Configura Rocket Enterprise Server su Amazon EC2 e distribuisci un'applicazione mainframe batch

Operazione	Description	Competenze richieste
Configura Rocket Enterprise Server e distribuisci un'applic azione demo.	Configura Rocket Enterpris e Server su Amazon EC2, quindi distribuisci l' BankDemo applicazione dimostrativa Rocket Software su Amazon. EC2 L' BankDemo applicazione è un'applicazione mainframe in batch che crea e quindi avvia l'output di stampa.	Architetto del cloud

Configura un server di stampa LRS su Amazon EC2

Operazione	Description	Competenze richieste
Ottieni una licenza del prodotto LRS per la stampa.	Per ottenere una licenza di prodotto LRS per LRSVPSX/ MFI, LRS/Queue, and LRS/ DIS, contattate il team di LRS Output Management. È necessario fornire i nomi	Costruisci piombo

Operazione	Description	Competenze richieste
	host delle EC2 istanze in cui verranno installati i prodotti LRS.	

Operazione	Description	Competenze richieste
Crea un'istanza Amazon EC2 Windows per installare LRS VPSX/MFI.	Avvia un'istanza Amazon EC2 Windows seguendo le istruzioni contenute in Launch an Amazon EC2 instance nella EC2 documentazione di Amazon. L'istanza deve soddisfare i seguenti requisiti hardware e software per LRS VPSX/MFI: • CPU: dual core • RAM — 16 GB • Unità: 500 GB • EC2 Istanza minima: m5.xlarge • Sistema operativo: Windows/Linux • Software: Internet Information Service (IIS) o Apache **Note** I requisiti hardware e software precedenti sono destinati a un piccolo parco stampanti (circa 500-1000). Per ottenere i requisiti completi, rivolgiti ai tuoi contatti LRS e AWS.	Architetto del cloud

Operazione	Description	Competenze richieste
	Quando crei l'istanza di Windows, procedi come segue: 1. Verificate che il nome EC2 host sia lo stesso usato per la licenza del prodotto LRS. 2. Abilita CGI in Amazon EC2 completando quanto segue: a. Connect alla tua EC2 istanza seguendo le istruzioni di	

Operazione	Description	Competenze richieste
Installa LRS VPSX/MFI sull' EC2 istanza.	1. Connettiti alla tua EC2 istanza seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. 2. **Note** Apri il link alla pagina di download del prodotto dall'e-mail di LRS che dovresti ricevere. : I prodotti LRS sono distribuiti tramite trasferim ento elettronico di file (EFT). 3. Scaricate LRS VPSX/MFI e decomprimete il file (cartella predefinita:). `c:\LRS` 4. Avviate LRS Product Installer dalla cartella decompressa per installare LRS VPSX/MFI. 5. Nel menu Seleziona funzionalità, selezionate VPSX® Server (V1R3.022), quindi scegliete Avanti per avviare il processo di installazione. Una volta completata l'installazione,	Architetto del cloud

Operazione	Description	Competenze richieste
	riceverai un messaggio di conferma.	
Installa LRS/Queue.	1. Connettiti alla tua EC2 istanza di Rocket Enterprise Server seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. 2. Apri il link alla pagina di download del prodotto LRS dall'e-mail di LRS che dovresti ricevere, scarica LRS/Queue, quindi decomprimi il file. 3. Vai alla posizione in cui hai scaricato i file, quindi avvia il programma di installazione del prodotto LRS per installare LRS/Queue.	Architetto del cloud

Operazione	Description	Competenze richieste
Installa LRS/DIS.	1. Connettiti alla tua VPSX/MFI EC2 istanza LRS seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. 2. Apri il link alla pagina di download del prodotto LRS dall'e-mail LRS che dovresti ricevere, scarica LRS/DIS, quindi decomprimi il file. 3. Vai alla posizione in cui hai scaricato i file, quindi avvia il programma di installazione del prodotto LRS. 4. In LRS Product Installer, espandete LRS Misc Tools, selezionate LRS DIS, quindi scegliete Avanti. 5. Seguite le altre istruzioni dell'LRS Product Installer per completare il processo di installazione.	Architetto del cloud

Operazione	Description	Competenze richieste
Create un gruppo target e registrate LRS VPSX/MFI EC2 come target.	Crea un gruppo target seguendo le istruzioni contenute in Crea un gruppo target per il tuo Network Load Balancer nella documentazione di Elastic Load Balancing. Quando crei il gruppo target, procedi come segue: 1. Nella pagina Specificare i dettagli del gruppo, per Scegli un tipo di destinazione, scegli Istanze. 2. Per Protocollo, scegli TCP. 3. Per Porta, scegli 5500. 4. Nella pagina Registra destinazioni, nella sezione Istanze disponibili, seleziona le istanze LRS VPSX/MFI EC2 .	Architetto del cloud

Operazione	Description	Competenze richieste
Crea un Network Load Balancer.	Segui le istruzioni contenute in Create a Network Load Balancer nella documenta zione di Elastic Load Balancing. Il Network Load Balancer indirizza il traffico da Rocket Enterprise Server a LRS. VPSX/MFI EC2 Quando create il Network Load Balancer, effettuate le seguenti operazioni nella pagina Listener and Routing: 1. Per Protocol (Protocollo), selezionare TCP. 2. Per Port, scegliete 5500. 3. Per Azione predefinita, scegli Inoltra a per il gruppo target che hai creato in precedenza.	Architetto del cloud

Integra Rocket Enterprise Server con LRS e LRS/Queue VPSX/MFI

Operazione	Description	Competenze richieste
Configura Rocket Enterpris e Server per LRS/Queue l'integrazione.	1. Connettiti alla tua EC2 istanza di Rocket Enterpris e Server seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon.	Architetto del cloud

Operazione	Description	Competenze richieste
	2. Nel menu Start di Windows, apri l'interfaccia utente di amministrazione di Rocket Enterprise Server. 3. Nella barra dei menu, scegli NATIVE. 4. Nel riquadro di navigazione, scegli Directory Server, quindi scegli BANKDEMO. 5. Da Generale nel riquadro di navigazione a sinistra, scorri verso il basso fino alla sezione Aggiuntivo per configurare le variabili di ambiente (LRSQ_ADDRESS, LRSQ_PORT, LRSQ_COMMAND) in modo che puntino a LRSQ. 6. Per LRSQ_ADDRESS, inserisci l'indirizzo IP o il nome DNS del Network Load Balancer che hai creato in precedenza. 7. Per LRSQ_PORT, inserite VPSX LRSQ Listener Port (5500). 8. Per LRSQ_COMMAND, inserite la posizione del percorso dell'eseguibile LRSQ.	

Operazione	Description	Competenze richieste
	Note Attualmente LRS supporta un limite massimo di 50 caratteri per i nomi DNS, ma questo limite è soggetto a modifiche in futuro. Se il tuo nome DNS è maggiore di 50, puoi utilizzare l'indirizzo IP del Network Load Balancer come alternativa.	

Operazione	Description	Competenze richieste
Configura Rocket Enterprise Server per l'integrazione con LRS. VPSX/MFI	1. Copia la `VPSX_MFI_R2` cartella dal programma di VPSX/MFI installazione di LRS nella posizione di Rocket Enterprise Server all'indirizzo. `C\BANKDEMO\print` 2. Connettiti alla tua EC2 istanza di Rocket Enterpris e Server seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. 3. Nel menu Start di Windows, apri l'interfaccia utente di amministrazione di Rocket Enterprise Server. 4. Nella barra dei menu, scegli NATIVE. 5. Nel riquadro di navigazio ne, scegli Directory Server, quindi scegli BANKDEMO. 6. In BANKDEMO, scegli JES. 7. In JES Program Path, aggiungi il `DLL(VPSX_MFI_R2)` percorso dalla posizione. `C\BANKDEMO\print`	Architetto cloud

Configura stampanti e utenti di stampa in Rocket Enterprise Server e LRS VPSX/MFI

Operazione	Description	Competenze richieste
Associate il modulo Rocket Software Print Exit al processo di esecuzione del server della stampante batch Rocket Enterprise Server.	1. Connettiti alla tua EC2 istanza di Rocket Enterprise Server seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. 2. Nel menu Start di Windows, apri l'interfaccia utente di amministrazione di Rocket Enterprise Server. 3. Nella barra dei menu, scegli NATIVE. 4. Nel riquadro di navigazione, scegli Directory Server, quindi scegli BANKDEMO. 5. In BANKDEMO, scegli JES e scorri verso il basso fino a Stampanti. 6. Nelle stampanti, associa il modulo Rocket Software Print Exit (per Batch) alla stampante LRSPRTE6 batch Rocket Enterprise Server Server Server Execution Process (SEP). Ciò consente il routing dell'output di stampa verso LRS VPSX/MFI.	Architetto del cloud

Operazione	Description	Competenze richieste
	7. Accedere all'interfaccia utente di Enterprise Server Administration. Per ulteriori informazioni sulla configurazione, vedete Using the Exit nella documentazione di Micro Focus.	

Operazione	Description	Competenze richieste
Aggiungi una stampante in LRS VPSX/MFI.	1. Connettiti alla tua VPSX/MFI EC2 istanza LRS seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. 2. Apri l'interfaccia Web VPSX dal menu Start di Windows. 3. Nel pannello di navigazione, scegliete Stampanti. 4. Scegli Aggiungi, quindi scegli Aggiungi stampante. 5. Nella pagina di configurazione della stampante, in Nome stampante, immettete Locale. 6. Per ID VPSX, immettere. VPS1 7. Per CommType, selezionate TCPIP/LRSQ. 8. Per Indirizzo host/IP, inserisci l'indirizzo IP della stampante fisica che desideri aggiungere. 9. Per Dispositivo, inserisci il nome del dispositivo. 10. Scegli Windows Driver o Linux/Mac Driver. 11. Scegliere Aggiungi.	Architetto del cloud

Operazione	Description	Competenze richieste
Crea un utente di stampa in LRS VPSX/MFI.	1. Connettiti alla tua VPSX/MFI EC2 istanza LRS seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. 2. Apri l'interfaccia Web VPSX dal menu Start di Windows. 3. Nel riquadro di navigazione, scegliete Sicurezza, quindi scegliete Utenti. 4. Nella colonna Nome utente, scegli admin, quindi scegli Copia. 5. Nella finestra Gestione del profilo utente, in Nome utente, inserisci un nome utente (ad esempio, PrintUser). 6. Per Descrizione, immettete una breve descrizione (ad esempio, Utente per la stampa di prova). 7. Scegliere Aggiorna. In questo modo viene creato un utente di stampa (ad esempio, PrintUser). 8. Nel riquadro di navigazione, in Utente, scegli il nuovo utente che hai creato. 9. Dal menu Comando, scegli Sicurezza.	Architetto del cloud

Operazione	Description	Competenze richieste
	10. Nella pagina Regole di sicurezza, scegli tutte le opzioni di sicurezza della stampante e del lavoro applicabili, quindi scegli Salva. 11. Per aggiungere il nuovo utente di stampa al gruppo Amministratore, vai al pannello di navigazione, scegli Sicurezza, quindi scegli Configura. 12. Nella finestra di configura zione della sicurezza, aggiungi il tuo nuovo utente di stampa alla colonna Amministratore.	

Configura l'autenticazione e l'autorizzazione di stampa

Operazione	Description	Competenze richieste
Crea un dominio AWS Managed Microsoft AD con utenti e gruppi.	1. Crea una Active Directory su AWS Managed Microsoft AD seguendo le istruzion i da Crea la tua directory AWS Managed Microsoft AD nella documentazione di AWS Directory Service. 2. Distribuisci un' EC2 istanza (Active Directory manager) e installa gli strumenti di Active Directory per gestire AWS Managed Microsoft	Architetto del cloud

Operazione	Description	Competenze richieste
	AD seguendo le istruzioni della Fase 3: Distribuisci un' EC2 istanza per gestire AWS Managed Microsoft AD nella documentazione di AWS Directory Service. 3. **Note** Connettiti alla tua EC2 istanza seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. : Quando ti connetti all' EC2 istanza, inserisci le tue credenziali di amministratore (per la directory che hai creato nel primo passaggio) nella finestra Sicurezza di Windows. 4. Nel menu Start di Windows, in Strumenti di amministrazione di Windows, scegli Utenti e computer di Active Directory. 5. Crea un utente di stampa nel dominio Active Directory	

Operazione	Description	Competenze richieste
	seguendo i passaggi della documentazione relativa alla creazione di un utente nella documentazione del servizio AWS Directory.	
Unisciti a LRS VPSX/MFI EC2 a un dominio AWS Managed Microsoft AD.	Aggiungi LRS VPSX/MFI EC2 al tuo dominio AWS Managed Microsoft AD automatic amente (documentazione AWS Knowledge Center) o manualmente (document azione AWS Directory Service).	Architetto del cloud

Operazione	Description	Competenze richieste
Configura e integra LRS/DIS con AWS Managed Microsoft AD.	1. Connettiti alla tua VPSX/MFI EC2 istanza LRS seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. 2. Nel menu Start di Windows, apri l'interfaccia Web VPSX. 3. Nel riquadro di navigazione, scegli Sicurezza, quindi scegli Configura. 4. Nella pagina Configurazione della sicurezza, nella sezione Parametri di sicurezza, per Tipo di sicurezza, seleziona Interno. 5. Inserisci le tue preferenze per le altre opzioni nella sezione Parametri di sicurezza. 6. Aprite la cartella LRS Output Management dal menu Start di Microsoft Windows, scegliete Server Start, quindi scegliete Server Stop. 7. Accedete a LRS VPSX/MFI con il nome utente e la password di Active Directory.	Architetto del cloud

Prova un flusso di lavoro di stampa

Operazione	Description	Competenze richieste
Avvia una richiesta di stampa in batch dall'app Rocket Software BankDemo .	1. Apri l'emulatore di terminale 3270 nell'istanza di Rocket Enterprise Server. EC2 2. Connect all' BankDemo app eseguendo il seguente comando: `connect 127.0.0.1:9278` 3. Nell'interfaccia della BankDemo riga di comando, per ID utente, inserisci B0001. Per Password, immettere una chiave non vuota. 4. Per l'opzione Richiedi dichiarazioni stampate, immettete X nella riga vuota. 5. Nella sezione Invia dichiarazione per, per Mail, inserisci Y, quindi premi F10.	Tecnico collaudatore
Controllate l'output di stampa in LRS VPSX/MFI.	1. Connettiti alla tua VPSX/MFI EC2 istanza LRS seguendo le istruzioni del Passaggio 2: Connettit i alla tua istanza nella EC2 documentazione di Amazon. 2. Nel menu Start di Windows, apri l'interfaccia Web VPSX.	Tecnico di test

Operazione	Description	Competenze richieste
	3. Nel riquadro di navigazione, scegliete Stampanti, quindi scegliete Output Queue. 4. Nella colonna Spool ID, scegliete l'ID bobina per la richiesta nella coda della stampante. 5. Nella scheda Azioni, nella colonna COMANDO, scegliete Sfoglia. È ora possibile visualizz are l'output di stampa di un estratto conto con le colonne relative al numero di conto. , Descrizione, data, importo e saldo. Per un esempio, vedete l'allegato batch_print_output per questo pattern.	

Risorse correlate

- Modernizzazione dell'output LRS (documentazione LRS)
- ANSI e controlli del trasporto delle macchine (documentazione IBM)
- Parole di comando del canale (documentazione IBM)
- Potenziamento dei carichi di lavoro mainframe aziendali su AWS con Micro Focus (blog AWS Partner Network)
- Crea una PAC Micro Focus Enterprise Server con Amazon EC2 Auto Scaling and Systems Manager (documentazione AWS Prescriptive Guidance)
- Flusso di dati AFP (Advanced Function Presentation) (documentazione IBM)
- Line Conditioned Data Stream (LCDS) (documentazione Compart)

Informazioni aggiuntive

Considerazioni

Durante il percorso di modernizzazione, potete prendere in considerazione un'ampia varietà di configurazioni sia per i processi batch mainframe che per l'output che generano. La piattaforma mainframe è stata personalizzata da ogni cliente e fornitore che la utilizza con requisiti particolari che influiscono direttamente sulla stampa. Ad esempio, la piattaforma attuale può incorporare IBM Advanced Function Presentation (AFP) o Xerox Line Condition Data Stream (LCDS) nel flusso di lavoro corrente. Inoltre, i caratteri di controllo del mainframe carriage e le parole di comando del canale possono influire sull'aspetto della pagina stampata e potrebbero richiedere una gestione speciale. Come parte del processo di pianificazione della modernizzazione, consigliamo di valutare e comprendere le configurazioni del proprio ambiente di stampa specifico.

Acquisizione dei dati di stampa

Rocket Software Print Exit trasmette le informazioni necessarie per consentire a LRS di VPSX/MFI elaborare efficacemente il file di spool. Le informazioni sono costituite da campi passati nei blocchi di controllo pertinenti, come:

- NOME DEL LAVORO
- PROPRIETARIO (USERID)
- DESTINAZIONE
- MODULO
- NOME DEL FILE
- SCRITTORE

LRS VPSX/MFI supporta i seguenti meccanismi batch mainframe per l'acquisizione di dati da Rocket Enterprise Server.

- Elaborazione COBOL in BATCH utilizzando istruzioni JCL print/spool SYSOUT standard z/OS DD/OUTPUT
- Elaborazione BATCH COBOL utilizzando istruzioni JCL print/spool CA-SPOOL SUBSYS DD standard z/OS
- IMS/COBOL print/spoolelaborazione tramite l'interfaccia CBLTDLI (per un elenco completo dei metodi supportati e degli esempi di programmazione, consultate la documentazione LRS inclusa nella licenza del prodotto).

Controlli dello stato del parco stampanti

LRS VPSX/MFI (LRS LoadX) è in grado di eseguire controlli approfonditi dello stato di salute, tra cui la gestione dei dispositivi e l'ottimizzazione operativa. La gestione dei dispositivi può rilevare guasti in un dispositivo di stampa e indirizzare la richiesta di stampa a una stampante funzionante. Per ulteriori informazioni sui controlli approfonditi dello stato delle flotte di stampanti, consultate la documentazione LRS inclusa nella licenza del prodotto.

Autenticazione e autorizzazione alla stampa

LRS/DIS enables LRS applications to authenticate user IDs and passwords by using Microsoft Active Directory or an LDAP server. In addition to basic print authorization, LRS/DISpuò anche applicare controlli di sicurezza di stampa a livello granulare nei seguenti casi d'uso:

- Gestisci chi può sfogliare il lavoro della stampante.
- Gestisci il livello di navigazione dei lavori di altri utenti.
- Gestisci le attività operative. Ad esempio, sicurezza a livello di comando come hold/release, purge, edit, copy e reindirizzamento. La sicurezza può essere impostata dall'ID utente o dal gruppo (simile al gruppo AD o al gruppo LDAP).

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Modernizzazione del mainframe: attiva DevOps AWS con Rocket Software Enterprise Suite

Kevin Yung, Amazon Web Services

Riepilogo

Sfide dei clienti

Organizations che eseguono applicazioni di base su hardware mainframe di solito incontrano alcune sfide quando l'hardware deve scalare per soddisfare le esigenze delle innovazioni digitali. Queste sfide includono i seguenti vincoli.

- Gli ambienti di sviluppo e test del mainframe non sono scalabili a causa della rigidità dei componenti hardware del mainframe e degli elevati costi di modifica.
- Lo sviluppo di mainframe sta affrontando una carenza di competenze, perché i nuovi sviluppatori non conoscono e non sono interessati ai tradizionali strumenti di sviluppo mainframe. Le tecnologie moderne come i container, le pipeline di integration/continuous distribuzione continua (CI/CD) e i moderni framework di test non sono disponibili nello sviluppo di mainframe.

Risultati del modello

Per affrontare queste sfide, Amazon Web Services (AWS) e Rocket Software Micro Focus, un partner AWS Partner Network (APN), hanno collaborato alla creazione di questo modello. La soluzione è progettata per aiutarti a raggiungere i seguenti risultati.

- Migliore produttività degli sviluppatori. Agli sviluppatori possono essere fornite nuove istanze di sviluppo mainframe in pochi minuti.
- Utilizzo di Cloud AWS per creare nuovi ambienti di test mainframe con capacità praticamente illimitata.
- Fornitura rapida di una nuova infrastruttura mainframe CI/CD . Il provisioning AWS può essere completato entro un'ora utilizzando e. AWS CloudFormation AWS Systems Manager
- Uso nativo di AWS DevOps strumenti per lo sviluppo di mainframe, tra cui AWS CodeBuild, AWS CodeCommit AWS CodePipeline AWS CodeDeploy, e Amazon Elastic Container Registry (Amazon ECR).
- Trasforma lo sviluppo tradizionale a cascata in uno sviluppo agile nei progetti mainframe.

Riepilogo delle tecnologie

In questo modello, lo stack di destinazione contiene i seguenti componenti.

Componenti logici	Soluzioni di implementazione	Description
Archivi di codice sorgente	AccuRev Server software Rocket CodeCommit, Amazon ECR	Gestione del codice sorgente: la soluzione utilizza due tipi di codice sorgente: • Codice sorgente del mainframe, ad esempio COBOL e JCL. • AWS modelli di infrastruttura e script di automazione Entrambi i tipi di codice sorgente richiedono il controllo della versione, ma vengono gestiti in modo diverso SCMs. Il codice sorgente distribuito nei server mainframe o Rocket Software Enterprise viene gestito in Rocket Software Micro Focus Server. AccuRev AWS i modelli e gli script di automazione sono gestiti in. CodeCommit Amazon ECR viene utilizzato per gli archivi di immagini Docker.
Istanze per sviluppatori aziendali	Amazon Elastic Compute Cloud (Amazon EC2), sviluppatore aziendale di software Rocket per Eclipse	Gli sviluppatori di mainframe possono sviluppare codice in Amazon EC2 utilizzando Rocket Software Enterpris e Developer for Eclipse. Ciò elimina la necessità di affidarsi

		all'hardware mainframe per scrivere e testare il codice.
Gestione delle licenze di Rocket Software Enterprise Suite	License Manager di Rocket Software Enterprise Suite	Per la gestione e la governanc e centralizzate delle licenze di Rocket Software Enterpris e Suite, la soluzione utilizza Rocket Software Enterprise Suite License Manager per ospitare la licenza richiesta.
Pipeline CI/CD	CodePipeline,, CodeBuild CodeDeploy, Rocket Software Enterprise Developer in un contenitore, Rocket Software Enterprise Test Server in un contenitore, Rocket Software Micro Focus Enterprise Server	I team di sviluppo del mainframe necessitano di CI/CD pipeline per eseguire la compilazione del codice, i test di integrazione e i test di regressione. In AWS, CodePipeline e CodeBuild può lavorare con Rocket Software Enterprise Developer ed Enterprise Test Server in un container in modo nativo.

Prerequisiti e limitazioni

Prerequisiti

Nome	Description
py3270	py3270 è un'interfaccia Python per x3270, un emulatore di terminale IBM 3270. Fornisce un'API per un sottoprocesso x3270 o s3270.
x3270	x3270 è un emulatore di terminale IBM 3270 per X Window System e Windows. Questo può essere usato dallo sviluppatore per il test delle unità a livello locale.

Robot-Framework-Mainframe-3270-Library	Mainframe3270 è una libreria per Robot Framework basata sul progetto py3270.
Rocket Software Verastream	Rocket Software Verastream è una piattaforma di integrazione che consente di testare gli asset mainframe nello stesso modo in cui vengono testate le app mobili, le applicazioni Web e i servizi Web SOA.
Programma di installazione e licenza Rocket Software Unified Functional Testing (UFT)	Rocket Software Unified Functional Testing è un software che fornisce l'automazione dei test funzionali e di regressione per applicazioni e ambienti software.
Programma di installazione e licenza di Rocket Software Enterprise Server	Enterprise Server fornisce l'ambiente di runtime per le applicazioni mainframe.
Programma di installazione e licenza di Rocket Software Enterprise Test Server	Rocket Software Enterprise Test Server è un ambiente di test delle applicazioni mainframe IBM.
Programma di AccuRev installazione e licenza Rocket Software per Server e programma di installazione e licenza Rocket Software Micro Focus per sistemi AccuRev operativi Windows e Linux	AccuRev fornisce la gestione del codice sorgente (SCM). Il AccuRev sistema è progettat o per essere utilizzato da un team di persone che stanno sviluppando un set di file.
Programma di installazione, patch e licenza di Rocket Software Enterprise Developer per Eclipse	Enterprise Developer fornisce agli sviluppatori di mainframe una piattaforma per sviluppare e mantenere le principali applicazioni mainframe online e in batch.

Limitazioni

- La creazione di un'immagine Windows Docker non è supportata in. CodeBuild Questo problema segnalato richiede il supporto dei team di Windows Kernel/HCS e Docker. La soluzione alternativa consiste nel creare un runbook di compilazione dell'immagine Docker utilizzando Systems

Manager. Questo modello utilizza la soluzione alternativa per creare immagini Rocket Software Enterprise Developer for Eclipse e Rocket Software Micro Focus Enterprise Test Server Container.

- La connettività del cloud privato virtuale (VPC) da non CodeBuild è ancora supportata in Windows, quindi il modello non utilizza Rocket Software License Manager per gestire le licenze nei contenitori Rocket Software Enterprise Developer e OpenText Rocket Software Enterprise Test Server.

Versioni del prodotto

- Rocket Software Enterprise Developer 5.5 o versione successiva
- Rocket Software Enterprise Test Server 5.5 o versione successiva
- Rocket Software Enterprise Server 5.5 o versione successiva
- Rocket Software 7.x o versione successiva AccuRev
- Immagine di base di Windows Docker per Rocket Software Enterprise Developer ed Enterprise Test Server: microsoft/dotnet-framework-4.7.2-runtime
- Immagine di base Linux Docker per AccuRev client: amazonlinux:2

Architecture

Ambiente mainframe

Nello sviluppo di mainframe convenzionali, gli sviluppatori devono utilizzare l'hardware mainframe per sviluppare e testare i programmi. Devono far fronte a limitazioni di capacità, ad esempio limitate per milioni di istruzioni al secondo (MIPS) per l' dev/test ambiente, e devono fare affidamento sugli strumenti disponibili sui computer mainframe.

In molte organizzazioni, lo sviluppo dei mainframe segue la metodologia di sviluppo a cascata, con i team che si affidano a cicli lunghi per rilasciare le modifiche. Questi cicli di rilascio sono generalmente più lunghi rispetto allo sviluppo di prodotti digitali.

Il diagramma seguente mostra più progetti mainframe che condividono hardware mainframe per il loro sviluppo. Nell'hardware mainframe, è costoso scalare un ambiente di sviluppo e test per più progetti.

Architettura AWS

Questo modello estende lo sviluppo del mainframe a. Cloud AWS Innanzitutto, utilizza AccuRev SCM per ospitare il codice sorgente del mainframe. AWS Quindi rende disponibili Enterprise Developer ed Enterprise Test Server per la creazione e il test del codice mainframe. AWS

Le sezioni seguenti descrivono i tre componenti principali del pattern.

1. SCM

Nel AWS, il pattern viene utilizzato AccuRev per creare un set di aree di lavoro SCM e il controllo della versione per il codice sorgente del mainframe. La sua architettura basata su stream consente lo sviluppo parallelo di mainframe per più team. Per unire una modifica, AccuRev utilizza il concetto di promozione. Per aggiungere tale modifica ad altre aree di lavoro, AccuRev utilizza il concetto di aggiornamento.

A livello di progetto, ogni team può creare uno o più flussi AccuRev per tenere traccia delle modifiche a livello di progetto. Questi sono chiamati flussi di progetto. Questi flussi di progetto vengono ereditati dallo stesso flusso principale. Il flusso principale viene utilizzato per unire le modifiche da diversi flussi di progetto.

Ogni stream di progetto può promuovere il codice verso AccuRev e viene impostato un trigger di promozione post per avviare la AWS pipeline CI/CD. La build riuscita di una modifica del flusso di progetto può essere promossa al flusso principale per ulteriori test di regressione.

Di solito, il flusso principale è chiamato flusso di integrazione del sistema. Quando viene effettuata una promozione da un flusso di progetto a un flusso di integrazione di sistema, un trigger successivo alla promozione avvia un'altra CI/CD pipeline per eseguire i test di regressione.

Oltre al codice mainframe, questo modello include AWS CloudFormation modelli, documenti Systems Manager Automation e script. Seguendo le infrastructure-as-code migliori pratiche, sono controllati in base alla versione. CodeCommit

Se è necessario sincronizzare il codice del mainframe con un ambiente mainframe per la distribuzione, Rocket Software fornisce la soluzione Enterprise Sync, che sincronizza il codice dall'SCM all'SCM del AccuRev mainframe.

2. Ambienti di sviluppo e test

In un'organizzazione di grandi dimensioni, scalare più di cento o addirittura più di mille sviluppatori mainframe è una sfida. Per risolvere questo vincolo, il modello utilizza istanze di Amazon EC2

Windows per lo sviluppo. Sulle istanze, sono installati gli strumenti Enterprise Developer for Eclipse. Lo sviluppatore può eseguire tutti i test e il debug del codice mainframe localmente sull'istanza.

AWS Systems Manager I documenti State Manager e Automation vengono utilizzati per automatizzare il provisioning delle istanze per sviluppatori. Il tempo medio per creare un'istanza per sviluppatori è di 15 minuti. Vengono preparati i seguenti software e configurazioni:

- AccuRev Client Windows per il check-out e il salvataggio del codice sorgente AccuRev
- Strumento Enterprise Developers for Eclipse, per la scrittura, il test e il debug del codice mainframe a livello locale
- Framework di test open source Python Behavior-driven development (BDD), framework di test Behave, py3270 e l'emulatore x3270 per la creazione di script per testare le applicazioni
- Uno strumento di sviluppo Docker per creare l'immagine Docker di Enterprise Test Server e testare l'applicazione nel contenitore Docker di Enterprise Test Server

Nel ciclo di sviluppo, gli sviluppatori utilizzano l' EC2 istanza per sviluppare e testare il codice mainframe a livello locale. Quando le modifiche locali vengono testate con successo, gli sviluppatori promuovono la modifica nel AccuRev server.

3. CI/CD oleodotti

Nello schema, le CI/CD pipeline vengono utilizzate per i test di integrazione e i test di regressione prima della distribuzione nell'ambiente di produzione.

Come spiegato nella sezione SCM, AccuRev utilizza due tipi di flussi: un flusso di progetto e un flusso di integrazione. Ogni stream è collegato a pipeline. CI/CD Per eseguire l'integrazione tra il AccuRev server e AWS CodePipeline, il pattern utilizza uno script di AccuRev post-promozione per creare un evento per avviare CI/CD.

Ad esempio, quando uno sviluppatore promuove una modifica a un flusso di progetto in AccuRev, avvia uno script di post-promozione da eseguire in Server. AccuRev Quindi lo script carica i metadati della modifica in un bucket Amazon Simple Storage Service (Amazon S3) per creare un evento Amazon S3. Questo evento avvierà l'esecuzione di una pipeline configurata. CodePipeline

Lo stesso meccanismo di avvio degli eventi viene utilizzato per il flusso di integrazione e le relative pipeline associate.

Nella CI/CD pipeline, CodePipeline si utilizza CodeBuild con il contenitore client AccuRev Linux per estrarre il codice più recente dagli stream. AccuRev Quindi la pipeline inizia CodeBuild a

utilizzare il contenitore Windows Enterprise Developer per compilare il codice sorgente e a utilizzare il contenitore Windows Enterprise Test Server per testare le applicazioni mainframe CodeBuild .

Le CI/CD pipeline vengono create utilizzando CloudFormation modelli e il blueprint verrà utilizzato per nuovi progetti. Utilizzando i modelli, un progetto impiega meno di un'ora per creare una nuova CI/CD pipeline. AWS

Per scalare la capacità di test del mainframe AWS, il modello crea la suite di DevOps test Rocket Software, Verastream e il server UFT. Utilizzando gli DevOps strumenti moderni, puoi eseguire tutti i test di cui hai bisogno. AWS

Nel diagramma seguente AWS è illustrato un esempio di ambiente di sviluppo mainframe con Rocket Software attivo.

Stack tecnologico Target

Questa sezione fornisce uno sguardo più da vicino all'architettura di ciascun componente del pattern.

1. Archivio del codice sorgente: SCM AccuRev

AccuRev SCM è configurato per gestire le versioni del codice sorgente mainframe. Per un'elevata disponibilità, AccuRev supporta le modalità primaria e di replica. Gli operatori possono eseguire il failover sulla replica durante la manutenzione sul nodo primario.

Per accelerare la risposta della CI/CD pipeline, il pattern utilizza Amazon CloudWatch Events per rilevare le modifiche al codice sorgente e avviare l'avvio della pipeline.

1. La pipeline è configurata per utilizzare una fonte Amazon S3.
2. Una regola CloudWatch Events è impostata per acquisire gli eventi S3 da un bucket S3 di origine.
3. La regola CloudWatch Events imposta un obiettivo per la pipeline.
4. AccuRev SCM è configurato per eseguire uno script di post-promozione a livello locale dopo il completamento della promozione.
5. AccuRev SCM genera un file XML che contiene i metadati della promozione e lo script carica il file XML nel bucket S3 di origine.
6. Dopo il caricamento, il bucket S3 di origine invia gli eventi che corrispondono alla regola Events e la regola CloudWatch Events avvia l'esecuzione della pipeline CloudWatch .

Quando la pipeline viene eseguita, avvia un CodeBuild progetto che prevede l'utilizzo di un contenitore client AccuRev Linux per estrarre il codice mainframe più recente da uno stream associato. AccuRev

Il diagramma seguente mostra una configurazione del server. AccuRev

2. Modello Enterprise Developer

Il modello utilizza i EC2 modelli Amazon per semplificare la creazione dell'istanza di sviluppo. Utilizzando State Manager, può applicare le impostazioni del software e della licenza alle EC2 istanze in modo coerente.

Il EC2 modello Amazon si basa sulle impostazioni del contesto VPC e sulle impostazioni predefinite dell'istanza e segue i requisiti di tagging aziendali. Utilizzando un modello, un team può creare le proprie nuove istanze di sviluppo.

All'avvio di un'istanza di sviluppo, tramite l'associazione ai tag, Systems Manager utilizza State Manager per applicare l'automazione. L'automazione include i seguenti passaggi generali.

1. Installa il software Enterprise Developer e installa le patch.
2. Installa il AccuRev client per Windows.
3. Installa lo script preconfigurato per consentire agli sviluppatori di partecipare allo AccuRev stream. Inizializza gli spazi di lavoro di Eclipse.
4. Installa strumenti di sviluppo, tra cui x3270, py3270 e Docker.
5. Configura le impostazioni della licenza in modo che punti a un bilanciamento del carico di License Manager.

Il diagramma seguente mostra un'istanza Enterprise Developer creata dal EC2 modello Amazon, con software e configurazione applicati all'istanza da State Manager. Le istanze Enterprise Developer si connettono AWS License Manager a per attivare la licenza.

3. Condutture CI/CD

Come spiegato nella sezione sull' AWS architettura, nel modello sono presenti pipeline a livello di progetto e CI/CD pipeline di integrazione di sistema. Ogni team di progetto mainframe crea una o più

CI/CD pipeline per creare i programmi che sta sviluppando in un progetto. Queste CI/CD pipeline di progetto controllano il codice sorgente da un flusso associato. AccuRev

In un team di progetto, gli sviluppatori promuovono il proprio codice nel AccuRev flusso associato. Quindi la promozione avvia la pipeline del progetto per creare il codice ed eseguire i test di integrazione.

Ogni CI/CD pipeline di progetto utilizza CodeBuild progetti con l'immagine Amazon ECR dello strumento Enterprise Developer e l'immagine Amazon ECR dello strumento Enterprise Test Server.

CodePipeline e CodeBuild vengono utilizzati per creare le pipeline. CI/CD Perché CodeBuild non CodePipeline avete commissioni o impegni anticipati, pagate solo per ciò che utilizzate. Rispetto all'hardware mainframe, la AWS soluzione riduce notevolmente i tempi di fornitura dell'hardware e abbassa i costi dell'ambiente di test.

Nello sviluppo moderno, vengono utilizzate più metodologie di test. Ad esempio, test-driven development (TDD), BDD e Robot Framework. Con questo modello, gli sviluppatori possono utilizzare questi strumenti moderni per i test dei mainframe. Ad esempio, utilizzando x3270, py3270 e lo strumento di test Python Behave, è possibile definire il comportamento di un'applicazione online. Puoi anche utilizzare il framework robotico build mainframe 3270 in queste pipeline. CI/CD

Il diagramma seguente mostra la pipeline del team stream. CI/CD

Il diagramma seguente mostra il rapporto di CI/CD test del progetto prodotto da CodePipeline in Mainframe3270 Robot Framework.

Il diagramma seguente mostra il rapporto di CI/CD test del progetto prodotto da CodePipeline in Py3270 e Behave BDD.

Dopo che i test a livello di progetto sono stati superati con successo, il codice testato viene promosso manualmente nel flusso di integrazione in SCM. AccuRev Puoi automatizzare questo passaggio dopo che i team avranno acquisito fiducia nella copertura dei test della loro pipeline di progetto.

Quando il codice viene promosso, la CI/CD pipeline di integrazione del sistema verifica il codice unito ed esegue test di regressione. Il codice unito viene promosso da tutti i flussi di progetto paralleli.

A seconda della granulometria richiesta dall'ambiente di test, i clienti possono disporre di più CI/CD pipeline di integrazione del sistema in un ambiente diverso, ad esempio UAT, Pre-Produzione.

Nello schema, gli strumenti utilizzati nella pipeline di integrazione del sistema sono Enterprise Test Server, UFT Server e Verastream. Tutti questi strumenti possono essere implementati nel contenitore Docker e utilizzati con. CodeBuild

Dopo aver testato con successo i programmi mainframe, l'elemento viene archiviato, con il controllo della versione, in un bucket S3.

Il diagramma seguente mostra una pipeline di integrazione del sistema. CI/CD

Dopo che l'artefatto è stato testato con successo nelle CI/CD pipeline di integrazione del sistema, può essere promosso per l'implementazione in produzione.

Se è necessario redistribuire il codice sorgente sul mainframe, Rocket Software offre la soluzione Enterprise Sync per sincronizzare il codice sorgente da Mainframe a Mainframe Endeavour. AccuRev

Il diagramma seguente mostra una pipeline di produzione che distribuisce l'artefatto in Enterprise Server CI/CD . In questo esempio, CodeDeploy orchestra la distribuzione dell'elemento mainframe testato in Enterprise Server.

Oltre alla descrizione dettagliata dell'architettura della CI/CD pipeline, consultate il post AWS DevOps sul blog Automatizzate migliaia di test mainframe AWS con Micro Focus Enterprise Suite per ulteriori informazioni sul test delle applicazioni mainframe in and. CodeBuild CodePipeline (Micro Focus è ora Rocket Software). Consultate il post del blog per le migliori pratiche e i dettagli su come eseguire i test del mainframe su. AWS

Tools (Strumenti)

AWS strumenti di automazione

- CloudFormation
- CloudWatch Eventi Amazon
- AWS CodeBuild
- AWS CodeDeploy
- AWS CodePipeline
- Amazon ECR

- Amazon S3
- Gestione dei segreti AWS
- AWS Systems Manager

Strumenti Rocket Software

- Sviluppatore Rocket Enterprise per Eclipse
- Server di test Rocket Enterprise
- Rocket Enterprise Server (distribuzione in produzione)
- Software Rocket AccuRev
- License Manager di Rocket Software Enterprise Suite
- Integratore di host Rocket Software Verastream
- Software Rocket UFT One

Altri strumenti

- x3270
- py3270
- Robot-Framework-Mainframe-3270-Libreria

Epiche

Crea l'infrastruttura AccuRev SCM

Operazione	Description	Competenze richieste
Implementa un server AccuRev SCM primario utilizzando. CloudFormation		AWS CloudFormation
Crea l'utente AccuRev amministratore.	Accedere a AccuRev SCM Server ed eseguire il comando CLI per creare un utente amministratore.	AccuRev Amministratore del server SCM

Operazione	Description	Competenze richieste
Crea AccuRev stream.	Crea AccuRev flussi che ereditano i flussi superiori in sequenza: Production, System Integration, Team Streams.	AccuRev Amministratore SCM
Crea gli account di AccuRev accesso per sviluppatori.	Utilizzate i comandi CLI AccuRev SCM per AccuRev creare account di accesso degli utenti per gli sviluppatori mainframe.	AccuRev Amministratore SCM

Crea il modello di EC2 lancio Amazon Enterprise Developer

Operazione	Description	Competenze richieste
Distribuisci il modello di EC2 lancio di Amazon utilizzando CloudFormation.	Utilizzalo CloudFormation per distribuire un modello di EC2 lancio Amazon per le istanze Enterprise Developer. Il modello include un documento Systems Manager Automation per l'istanza Rocket Enterprise Developer.	AWS CloudFormation
Crea l'istanza Enterprise Developer dal EC2 modello Amazon.		Accesso alla console AWS e competenze per sviluppatori mainframe

Crea l'immagine Docker dello strumento Enterprise Developer

Operazione	Description	Competenze richieste
Crea l'immagine Docker dello strumento Enterprise Developer.	Usa il comando Docker e lo strumento Enterprise Developer Dockerfile per creare l'immagine Docker.	Docker
Crea il repository Docker in Amazon ECR.	Sulla console Amazon ECR, crea il repository per l'immagine Enterprise Developer Docker.	Amazon ECR
Invia l'immagine Docker dello strumento Enterprise Developer ad Amazon ECR.	Esegui il comando Docker push per inviare l'immagine Docker dello strumento Enterprise Developer per salvarla nel repository Docker in Amazon ECR.	Docker

Crea l'immagine Docker di Enterprise Test Server

Operazione	Description	Competenze richieste
Crea l'immagine Docker di Enterprise Test Server.	Usa il comando Docker e il Dockerfile di Enterprise Test Server per creare l'immagine Docker.	Docker
Crea il repository Docker in Amazon ECR.	Sulla console Amazon ECR, crea l'archivio Amazon ECR per l'immagine Docker di Enterprise Test Server.	Amazon ECR

Operazione	Description	Competenze richieste
Invia l'immagine Docker di Enterprise Test Server ad Amazon ECR.	Esegui il comando Docker push per inviare e salvare l'immagine Docker di Enterpris e Test Server in Amazon ECR.	Docker

Crea la pipeline di stream del team CI/CD

Operazione	Description	Competenze richieste
Crea il CodeCommit repositor y.	Sulla CodeCommit console, crea un repository basato su Git per l'infrastruttura e il codice. CloudFormation	AWS CodeCommit
Carica il CloudFormation modello e il codice di automazione nel CodeCommit repository.	Esegui il comando Git push per caricare il CloudForm ation modello e il codice di automazione nel repository.	Git
Implementa la CI/CD pipeline di stream del team utilizzando. CloudFormation	Utilizza il CloudFormation modello preparato per distribui re una pipeline di flusso del team. CI/CD	AWS CloudFormation

Crea la CI/CD pipeline di integrazione del sistema

Operazione	Description	Competenze richieste
Crea l'immagine UFT Docker.	Usa il comando Docker e il Dockerfile UFT per creare l'immagine Docker.	Docker

Operazione	Description	Competenze richieste
Crea il repository Docker in Amazon ECR per l'immagine UFT.	Sulla console Amazon ECR, crea il repository Docker per l'immagine UFT.	Amazon ECR
Invia l'immagine UFT Docker ad Amazon ECR.	Esegui il comando Docker push per inviare e salvare l'immagine Docker di Enterpris e Test Server in Amazon ECR.	Docker
Crea l'immagine Verastream Docker.	Usa il comando Docker e il Dockerfile Verastream per creare l'immagine Docker.	Docker
Crea il repository Docker in Amazon ECR per l'immagine Verastream.	Sulla console Amazon ECR, crea il repository Docker per l'immagine Verastream.	Amazon ECR
Implementa la pipeline di integrazione del sistema utilizzando. CI/CD CloudForm ation	Utilizza il CloudFormation modello preparato per implementare una pipeline di integrazione del sistema. CI/ CD	AWS CloudFormation

Crea una CI/CD pipeline di distribuzione di produzione

Operazione	Description	Competenze richieste
Implementa Enterprise Server utilizzando AWS Quick Start.	Per distribuire Enterprise Server utilizzando CloudForm ation, avviare Enterprise Server su AWS Quick Start.	AWS CloudFormation
Implementa una CI/CD pipeline di distribuzione in produzione.	Sulla CloudFormation console, utilizza il CloudFormation modello per implementare	AWS CloudFormation

Operazione	Description	Competenze richieste
	una pipeline di distribuzione in produzione. CI/CD	

Risorse correlate

Riferimenti

- AWS DevOps Blog - Automatizza migliaia di test del mainframe AWS con Micro Focus Enterprise Suite (Micro Focus è ora Rocket Software).
- repository py3270/py3270 GitHub
- Archivio della libreria Altran-PT-GDC/Robot-Framework-Mainframe-3270 GitHub
- Benvenuti a comportarvi bene!
- Blog dei partner APN - Tag: Micro Focus (Micro Focus è ora Rocket Software).
- Avvio di un'istanza da un modello di avvio

Marketplace AWS

- Rocket Software UFT One

AWS Quick Start

- Rocket Enterprise Server attivo AWS

Modernizza i carichi di lavoro di stampa online mainframe su AWS utilizzando Micro Focus Enterprise Server e LRS VPSX/MFI

Shubham Roy e Kevin Yung, Amazon Web Services

Abraham Rondon, Micro Focus

Guy Tucker, Levi, Ray and Shoup Inc

Riepilogo

Questo modello mostra come modernizzare i carichi di lavoro di stampa online mainframe critici per l'azienda sul cloud Amazon Web Services (AWS) utilizzando Micro Focus Enterprise Server come runtime per un'applicazione mainframe modernizzata e LRS VPSX/MFI (Micro Focus Interface) come server di stampa. Il modello si basa sull'approccio di modernizzazione del mainframe replatform. In questo approccio, esegui la migrazione della tua applicazione mainframe online su Amazon Elastic Compute Cloud (Amazon EC2) e la migrazione del database mainframe, ad esempio IBM for DB2) z/OS, to Amazon Relational Database Service (Amazon RDS). The authentication and authorization for the modernized print workflow is performed by AWS Directory Service for Microsoft Active Directory, also known as AWS Managed Microsoft AD. The LRS Directory Information Server (LRS/DIS è integrato con AWS Managed Microsoft AD per l'autenticazione e l'autorizzazione del flusso di lavoro di stampa. Modernizzando i carichi di lavoro di stampa online, puoi ridurre i costi dell'infrastruttura IT, mitigare il debito tecnico legato alla manutenzione dei sistemi legacy, rimuovere i silos di dati, aumentare l'agilità e l'efficienza con un DevOps modello e sfruttare le risorse e l'automazione su richiesta nel cloud AWS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un carico di lavoro di stampa o gestione dell'output online su mainframe
- Conoscenze di base su come ricostruire e fornire un'applicazione mainframe eseguibile su Micro Focus Enterprise Server (per ulteriori informazioni, consultate la scheda tecnica di Enterprise Server nella documentazione Micro Focus).
- Conoscenza di base delle soluzioni e dei concetti di stampa su cloud di LRS (per ulteriori informazioni, vedete Output Modernization nella documentazione di LRS).

- Software e licenza Micro Focus Enterprise Server (per ulteriori informazioni, contattate il reparto vendite di Micro Focus).
- VPSX/MFI, LRS/Queue, and LRS/DISSoftware e licenze LRS (per ulteriori informazioni, contattate il reparto vendite LRS).

> **Note**
>
> Per ulteriori informazioni sulle considerazioni sulla configurazione per i carichi di lavoro di stampa mainframe online, vedete Considerazioni nella sezione Informazioni aggiuntive di questo modello.

Versioni del prodotto

- Micro Focus Enterprise Server 8.0 o versioni successive
- LRS VPSX/MFI V1R3 o versione successiva

Architecture

Stack tecnologico di origine

- Sistema operativo: IBM z/OS
- Linguaggio di programmazione: Common Business-Oriented Language (COBOL) e Customer Information Control System (CICS)
- Database: IBM DB2 per z/OS IBM Information Management System (IMS) e Virtual Storage Access Method (VSAM)
- Sicurezza: Resource Access Control Facility (RACF), CA Top Secret for z/OS e Access Control Facility 2 () ACF2
- Gestione della stampa e dell'output: prodotti di z/OS stampa mainframe IBM (IBM Infoprint Server for z/OS, LRS e CA View)

Stack tecnologico Target

- Sistema operativo: Microsoft Windows Server in esecuzione su Amazon EC2
- Elaborazione — Amazon EC2

- Linguaggio di programmazione: COBOL e CICS
- Banca dati — Amazon RDS
- Sicurezza: AWS Managed Microsoft AD
- Gestione della stampa e dell'output: soluzione di stampa LRS su AWS
- Ambiente di runtime mainframe: Micro Focus Enterprise Server

Architettura di origine

Il diagramma seguente mostra una tipica architettura allo stato attuale per un carico di lavoro di stampa online mainframe.

Il diagramma mostra il flusso di lavoro seguente:

1. Gli utenti eseguono transazioni commerciali su un sistema di coinvolgimento (SoE) basato su un'applicazione IBM CICS scritta in COBOL.
2. Il SoE richiama il servizio mainframe, che registra i dati delle transazioni commerciali in un database system-of-records (SoR) come IBM for z/OS. DB2
3. Il SoR conserva i dati aziendali del SoE.
4. Un utente avvia una richiesta per generare un output di stampa dal CICS SoE, che avvia un'applicazione di transazione di stampa per elaborare la richiesta di stampa.
5. L'applicazione per le transazioni di stampa (ad esempio un programma CICS e COBOL) estrae i dati dal database, li formatta in base ai requisiti aziendali e genera risultati aziendali (dati di stampa) come estratti conto, carte d'identità o estratti conto di prestito. Quindi, l'applicazione invia una richiesta di stampa utilizzando il metodo di accesso virtuale alle telecomunicazioni (VTAM). Un server di z/OS stampa (come IBM Infoprint Server) utilizza NetSpool un componente VTAM simile per intercettare le richieste di stampa, quindi crea set di dati di output di stampa sulla bobina JES utilizzando i parametri di output JES. I parametri di output JES specificano le informazioni di routing utilizzate dal server di stampa per trasmettere l'output a una particolare stampante di rete. Il termine VTAM si riferisce al z/OS Communications Server e all'elemento dei servizi SNA (System Network Architecture) di z/OS.
6. Il componente di trasmissione dell'output di stampa trasmette i set di dati di stampa in uscita dalla bobina JES a stampanti o server di stampa remoti, come LRS (come illustrato in questo modello), IBM Infoprint Server o destinazioni e-mail.

Architettura Target

Il diagramma seguente mostra un'architettura per un carico di lavoro di stampa online mainframe distribuito nel cloud AWS:

Il diagramma mostra il flusso di lavoro seguente:

1. Un utente avvia una richiesta di stampa da un'interfaccia utente online (CICS) per creare output di stampa, come estratti conto di fatturazione, carte d'identità o estratti conto di prestito.
2. L'applicazione online mainframe (riplatformata su Amazon EC2) utilizza il runtime Micro Focus Enterprise Server per estrarre i dati dal database dell'applicazione, applicare la logica aziendale ai dati, formattare i dati e quindi inviarli a una destinazione di stampa utilizzando Micro Focus CICS Print Exit (DFHUPRNT).
3. Il database dell'applicazione (un SoR eseguito su Amazon RDS) mantiene i dati per l'output di stampa.
4. La soluzione di VPSX/MFI stampa LRS è implementata su Amazon EC2 e i suoi dati operativi sono archiviati in Amazon Elastic Block Store (Amazon EBS). LRS VPSX/MFI utilizza un nome di coda. TCP/IP-based LRS/Queue transmission agent to collect print data through the Micro Focus CICS Print Exit API (DFHUPRNT) and deliver the data to a specified printer destination. The original TERMID (TERM) that’s used in the modernized CICS application is used as the VPSX/MFI

> **Note**
>
> La soluzione di destinazione in genere non richiede modifiche alle applicazioni per adattarsi ai linguaggi di formattazione del mainframe, come IBM Advanced Function Presentation (AFP) o Xerox Line Condition Data Stream (LCDS). Per ulteriori informazioni sull'utilizzo di Micro Focus per la migrazione e la modernizzazione delle applicazioni mainframe su AWS, consulta Empowering Enterprise Mainframe Workloads on AWS with Micro Focus nella documentazione AWS.

Architettura dell'infrastruttura AWS

Il diagramma seguente mostra un'architettura di infrastruttura AWS altamente disponibile e sicura per un carico di lavoro di stampa online mainframe:

Il diagramma mostra il flusso di lavoro seguente:

1. L'applicazione online mainframe (scritta in un linguaggio di programmazione come CICS o COBOL) utilizza la logica aziendale principale per elaborare e generare output di stampa, come estratti conto, carte d'identità e dichiarazioni di prestito. L'applicazione online è distribuita su Amazon EC2 in due zone di disponibilità (AZ) per l'alta disponibilità (HA) e utilizza Micro Focus CICS Print Exit per indirizzare l'output di stampa a LRS VPSX/MFI per la stampa da parte dell'utente finale.
2. LRS utilizza un comando. VPSX/MFI TCP/IP-based LRS/Queue transmission agent to collect or capture print data from the Micro Focus online Print Exit programming interface. Online Print Exit passes the necessary information to enable LRS VPSX/MFI to effectively process the print file and dynamically build LRS/Queue

   > **Note**
   >
   > Per ulteriori informazioni sui vari metodi di programmazione delle applicazioni CICS per la stampa e su come sono supportati nel server Micro Focus Enterprise e LRS VPSX/MFI, vedete Print data capture nella sezione Informazioni aggiuntive di questo modello.

3.
   > **Note**
   >
   > Un Network Load Balancer fornisce un nome DNS per integrare Micro Focus Enterprise Server con LRS VPSX/MFI. : LRS VPSX/MFI supports a Layer 4 load balancer. The Network Load Balancer also does a basic health check on LRS VPSX/MFI e indirizza il traffico verso le destinazioni registrate che sono integre.

4. Il server di VPSX/MFI stampa LRS è distribuito EC2 su Amazon in due zone di disponibilità per HA e utilizza Amazon EBS come archivio dati operativo. LRS VPSX/MFI supporta le modalità di servizio attivo-attivo e attivo-passivo. Questa architettura utilizza più zone di disponibilità in una coppia attiva-passiva come standby attivo e hot standby. Il Network Load Balancer esegue un controllo dello stato delle istanze LRS e indirizza il traffico verso VPSX/MFI EC2 istanze hot standby in un'altra zona di disponibilità se un'istanza attiva non è integra. Le richieste di stampa vengono mantenute nella LRS Job Queue localmente in ciascuna istanza. EC2 In caso di ripristino, è necessario riavviare un'istanza fallita affinché i servizi LRS riprendano l'elaborazione della richiesta di stampa.

> **Note**
>
> LRS VPSX/MFI può anche eseguire controlli di integrità a livello di parco stampanti. Per ulteriori informazioni, vedete Controlli dello stato del parco stampanti nella sezione Informazioni aggiuntive di questo modello.

5. AWS Managed Microsoft AD si integra con LRS/DIS per eseguire l'autenticazione e l'autorizzazione del flusso di lavoro di stampa. Per ulteriori informazioni, consulta Autenticazione e autorizzazione alla stampa nella sezione Informazioni aggiuntive di questo modello.
6. LRS VPSX/MFI utilizza Amazon EBS per lo storage a blocchi. Puoi eseguire il backup dei dati di Amazon EBS da EC2 istanze attive su Amazon S3 point-in-time come snapshot e ripristinarli su volumi EBS in hot standby. Per automatizzare la creazione, la conservazione e l'eliminazione degli snapshot di volume Amazon EBS, puoi utilizzare Amazon Data Lifecycle Manager per impostare la frequenza degli snapshot automatici e ripristinarli in base ai requisiti RTO/RPO.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Block Store (Amazon EBS) fornisce volumi di storage a livello di blocco da utilizzare con le istanze Amazon. EC2 Il comportamento dei volumi EBS è simile a quello dei dispositivi a blocchi non formattati e non elaborati. Puoi montare questi volumi come dispositivi sulle istanze.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS.
- AWS Directory Service per Microsoft Active Directory (AD), noto anche come AWS Managed Microsoft Active Directory, consente ai carichi di lavoro compatibili con le directory e alle risorse AWS di utilizzare Active Directory gestita in AWS.

Altri strumenti

- LRS VPSX/MFI (Micro Focus Interface), sviluppato congiuntamente da LRS e Micro Focus, acquisisce l'output da una bobina JES di Micro Focus Enterprise Server e lo consegna in modo affidabile a una destinazione di stampa specificata.
- LRS Directory Information Server (LRS/DIS) viene utilizzato per l'autenticazione e l'autorizzazione durante il flusso di lavoro di stampa.
- LRS/Queue is a TCP/IP-based LRS/Queue transmission agent, used by LRS VPSX/MFI, per raccogliere o acquisire dati di stampa tramite l'interfaccia di programmazione Print Exit online di Micro Focus.
- Micro Focus Enterprise Server è un ambiente di distribuzione delle applicazioni per applicazioni mainframe. Fornisce l'ambiente di esecuzione per le applicazioni mainframe che vengono migrate o create utilizzando qualsiasi versione di Micro Focus Enterprise Developer.

Epiche

Configura Micro Focus Enterprise Server su Amazon EC2 e distribuisci un'applicazione mainframe online

Operazione	Description	Competenze richieste
Configurate Micro Focus Enterprise Server e distribuite un'applicazione demo online.	Configurate Micro Focus Enterprise Server su Amazon EC2, quindi distribuite l'applicazione Micro Focus Account Demo (ACCT Demo) su Amazon EC2 seguendo le istruzioni di Tutorial: CICS Support nella documentazione Micro Focus. L'applicazione ACCT Demo è un'applicazione mainframe online (CICS) che crea e quindi avvia l'output di stampa.	Architetto del cloud

Configura un server di stampa LRS su Amazon EC2

Operazione	Description	Competenze richieste
Ottieni una licenza del prodotto LRS per la stampa.	Per ottenere una licenza di prodotto LRS per LRSVPSX/MFI, LRS/Queue, and LRS/DIS, contattate il team di LRS Output Management. È necessario fornire i nomi host delle EC2 istanze in cui verranno installati i prodotti LRS.	Costruisci piombo
Crea un'istanza Amazon EC2 Windows per installare LRS VPSX/MFI.	Avvia un'istanza Amazon EC2 Windows seguendo le istruzioni del Passaggio 1: Avvia un'istanza nella EC2 documentazione di Amazon. L'istanza deve soddisfare i seguenti requisiti hardware e software per LRS VPSX/MFI: • CPU: dual core • RAM — 16 GB • Unità: 500 GB • EC2 Istanza minima: m5.xlarge • Sistema operativo: Windows/Linux • Software: Internet Informati on Service (IIS) o Apache	Architetto del cloud

Operazione	Description	Competenze richieste
	Note I requisiti hardware e software precedent i sono destinati a un piccolo parco stampanti (circa 500-1000). Per ottenere i requisiti completi, rivolgiti ai tuoi contatti LRS e AWS. Quando crei l'istanza di Windows, procedi come segue: 1. Verificate che il nome EC2 host sia lo stesso usato per la licenza del prodotto LRS. 2. Abilita CGI in Amazon EC2 completando quanto segue: a. Connettiti alla tua EC2 istanza seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. b. Nel menu Start di Windows, trova e apri Server Manager.	

Operazione	Description	Competenze richieste
	c. In Server Manager, scegli Dashboard, Quick Start, Aggiungi ruoli e funzionalità. Quindi, scegli Ruoli del server. d. In Ruoli server, scegli WebServer (IIS), quindi scegli Sviluppo applicazioni. e. In Sviluppo di applicazioni, seleziona la casella di controllo CGI. f. Segui le istruzioni della procedura guidata di aggiunta di ruoli e funzionalità di Windows Server Manager per installare CGI. g. Aprire la porta 5500 nel firewall di Windows dell' EC2 istanza per la comunicazione. LRS/Queue	

Operazione	Description	Competenze richieste
Installa LRS VPSX/MFI sull' EC2 istanza.	1. Connettiti alla tua EC2 istanza seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. 2. **Note** Apri il link alla pagina di download del prodotto dall'e-mail di LRS che dovresti ricevere. : I prodotti LRS sono distribuiti tramite trasferim ento elettronico di file (EFT). 3. Scaricate LRS VPSX/MFI e decomprimete il file (cartella predefinita:). `c:\LRS` 4. Avviate LRS Product Installer dalla cartella decompressa per installare LRS VPSX/MFI. 5. Nel menu Seleziona funzionalità, selezionate VPSX® Server (V1R3.022), quindi scegliete Avanti per avviare il processo di installazione. Una volta completata l'installazione,	Architetto del cloud

Operazione	Description	Competenze richieste
	riceverai un messaggio di conferma.	
Installa LRS/Queue.	1. Connettiti alla tua EC2 istanza Micro Focus Enterprise Server seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. 2. Aprite il link alla pagina di download del prodotto LRS dall'e-mail LRS che dovreste ricevere, scaricate LRS/Queue e decomprim ete il file. 3. Vai alla posizione in cui hai scaricato i file, quindi avvia il programma di installaz ione del prodotto LRS per installare LRS/Queue.	Architetto del cloud

Operazione	Description	Competenze richieste
Installa LRS/DIS.	1. Connettiti alla tua VPSX/MFI EC2 istanza LRS seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. 2. Apri il link alla pagina di download del prodotto LRS dall'e-mail LRS che dovresti ricevere, scarica LRS/DIS, quindi decomprimi il file. 3. Vai alla posizione in cui hai scaricato i file, quindi avvia il programma di installazione del prodotto LRS. 4. In LRS Product Installer, espandete LRS Misc Tools, selezionate LRS DIS, quindi scegliete Avanti. 5. Seguite le altre istruzioni dell'LRS Product Installer per completare il processo di installazione.	Architetto del cloud

Operazione	Description	Competenze richieste
Create un gruppo target e registrate LRS VPSX/MFI EC2 come target.	Crea un gruppo target seguendo le istruzioni contenute in Crea un gruppo target per il tuo Network Load Balancer nella documentazione di Elastic Load Balancing. Quando crei il gruppo target, procedi come segue: 1. Nella pagina Specificare i dettagli del gruppo, per Scegli un tipo di destinazione, scegli Istanze. 2. Per Protocollo, scegli TCP. 3. Per Porta, scegli 5500. 4. Nella pagina Registra destinazioni, nella sezione Istanze disponibili, seleziona le istanze LRS VPSX/MFI EC2 .	Architetto del cloud

Operazione	Description	Competenze richieste
Crea un Network Load Balancer.	Segui le istruzioni contenute in Create a Network Load Balancer nella documenta zione di Elastic Load Balancing. Il Network Load Balancer indirizza il traffico da Micro Focus Enterprise Server a LRS. VPSX/MFI EC2 Quando create il Network Load Balancer, effettuate le seguenti operazioni nella pagina Listener and Routing: 1. Per Protocol (Protocollo), selezionare TCP. 2. Per Port, scegliete 5500. 3. Per Azione predefinita, scegli Inoltra a per il gruppo target che hai creato in precedenza.	Architetto del cloud

Integra Micro Focus Enterprise Server con LRS VPSX/MFI e LRS/Queue

Operazione	Description	Competenze richieste
Configurate Micro Focus Enterprise Server per LRS/ Queue l'integrazione.	1. Connettiti alla tua EC2 istanza Micro Focus Enterprise Server seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon.	Architetto del cloud

Operazione	Description	Competenze richieste
	2. Nel menu Start di Windows, aprite l'interfaccia utente di amministrazione di Micro Focus Enterprise Server. 3. Nella barra dei menu, scegliete NATIVE. 4. Nel pannello di navigazione, scegli Directory Server, quindi scegli BANKDEMO o la regione del tuo server Enterprise. 5. Da Generale, nel riquadro di navigazione a sinistra, scorri verso il basso fino alla sezione Aggiuntivo per configurare le variabili di ambiente (LRSQ_ADDRESS, LRSQ_PORT, LRSQ_COMMAND) in modo che puntino a LRSQ. 6. Per LRSQ_ADDRESS, inserisci l'indirizzo IP o il nome DNS del Network Load Balancer creato in precedenza. 7. Per LRSQ_PORT, inserite VPSX LRSQ Listener Port (5500). 8. Per LRSQ_COMMAND, inserite la posizione del percorso dell'eseguibile LRSQ.	

Operazione	Description	Competenze richieste
	9. **Note** Attualmente LRS supporta un limite massimo di 50 caratteri per i nomi DNS, ma questo limite è soggetto a modifiche in futuro. Se il tuo nome DNS è maggiore di 50, puoi utilizzare l'indirizzo IP del Network Load Balancer come alternativa.	

Operazione	Description	Competenze richieste
Rendete CICS Print Exit (DFHUPRNT) disponibile per l'inizializzazione di Micro Focus Enterprise Server.	1. Connettiti alla tua EC2 istanza Micro Focus Enterprise Server seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. 2. **Note** Copiate CICS Print Exit (DFHUPRNT) dalla cartella VPSX/MFI eseguibile LRS (denomina taVPSX_MFI_R2) nella posizione dell'istanza di Micro Focus Enterprise Server. EC2 Per i sistemi a 32 bit, la posizione è. C:\Program Files (x86) \Micro Focus \Enterprise Server\bin Per i sistemi a 64 bit, la posizione èC:\Program Files (x86) \Micro Focus \Enterprise Server\bin64 .: Il DFHUPRNT_	Architetto del cloud

Operazione	Description	Competenze richieste
	`64.dll` file deve essere rinominato `DFHUPRNT.dll` quando viene copiato. Verificare che Micro Focus Enterprise Server abbia rilevato CICS Print Exit (DFHUPRNT) 1. Arrestate e avviate Micro Focus Enterprise Server. 2. Nel pannello di amministrazione di Micro Focus Enterprise Server, aprire Monitor, Logs, Console logs. 3. Controllate i registri della console per il seguente messaggio: «3270 printer user exit DFHUPRNT installato correttamente».	

Operazione	Description	Competenze richieste
Definite l'ID del terminale della stampante CICS (TERMIDs) come Micro Focus Enterprise Server.	Abilitate la stampa 3270 in Micro Focus Enterprise Server 1. Nel pannello di amministrazione di Micro Focus Enterprise Server, aprire CICS, Resources, By Group. 2. Dal pannello di navigazione a sinistra, scegliete SIT (System Initialization Table), quindi scegliete BNKCICV. 3. Nella sezione Generale, scorri verso il basso fino a 3270, quindi seleziona la casella di controllo 3270 Stampa. Definite il terminale della stampante CICS in Micro Focus Enterprise Server 1. Nel pannello di amministrazione di Micro Focus Enterprise Server, aprire CICS, Resources, By Type. 2. Dal pannello di navigazione a sinistra, scegliete Termine, quindi scegliete Nuovo. Si apre il modulo Create Terminal Resource. 3. In Nome, inserite il nome della coda di stampa LRS.	Architetto del cloud

Operazione	Description	Competenze richieste
	(Nota: questo modello utilizza «P275" come ID terminale della stampante CICS e coda di stampa LRS VPSX.) 4. Per Gruppo, inserisci BANKTERM. 5. Per Auto Install — Model, inserisci NO. 6. Per Identificatori di terminale - Tipo di terminale , immettere DFHPRT32. 7. Per Net name, immettere VTAMP275. 8. Per Terminal Usage, selezionate la casella di controllo In Service. 9. Scorri nella parte superiore della pagina, quindi scegli Salva. 10. Scegli Installa. Un messaggio pop-up mostra un messaggio di installaz ione riuscita.	

Configurate stampanti e utenti di stampa in Micro Focus Enterprise Server e LRS VPSX/MFI

Operazione	Description	Competenze richieste
Crea una coda di stampa in LRS VPSX.	1. Connettiti alla tua VPSX/MFI EC2 istanza LRS seguendo le istruzioni del Passaggio 2: Connettit	Architetto del cloud

Operazione	Description	Competenze richieste
	i alla tua istanza nella EC2 documentazione di Amazon. 2. Apri l'interfaccia Web VPSX dal menu Start di Windows. 3. Nel pannello di navigazione, scegliete Stampanti. 4. Scegli Aggiungi, quindi scegli Aggiungi stampante. 5. Nella pagina di configura zione della stampante , per Nome stampante, immettete P275. 6. Per ID VPSX, immettere. VPS1 7. Per CommType, seleziona te TCPIP/LRSQ. 8. Per Indirizzo host/IP, inserisci l'indirizzo IP della stampante fisica che desideri aggiungere. 9. Per Dispositivo, inserisci il nome del dispositivo. 10. Scegli Windows Driver o Linux/Mac Driver. 11. Scegliere Aggiungi. **Note** La coda di stampa deve essere equivalen te alla stampa	

Operazione	Description	Competenze richieste
	TERMIDs creata in Micro Focus Enterpris e Server.	

Operazione	Description	Competenze richieste
Crea un utente di stampa in LRS VPSX/MFI.	1. Connettiti alla tua VPSX/MFI EC2 istanza LRS seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. 2. Apri l'interfaccia Web VPSX dal menu Start di Windows. 3. Nel riquadro di navigazione, scegliete Sicurezza, quindi scegliete Utenti. 4. Nella colonna Nome utente, scegli admin, quindi scegli Copia. 5. Nella finestra Gestione del profilo utente, in Nome utente, inserisci un nome utente (ad esempio, PrintUser). 6. Per Descrizione, immettete una breve descrizione (ad esempio, Utente per la stampa di prova). 7. Scegliere Aggiorna. In questo modo viene creato un utente di stampa (ad esempio, PrintUser). 8. Nel riquadro di navigazione, in Utente, scegli il nuovo utente che hai creato. 9. Dal menu Comando, scegli Sicurezza.	Architetto del cloud

Operazione	Description	Competenze richieste
	10. Nella pagina Regole di sicurezza, scegli tutte le opzioni di sicurezza della stampante e del lavoro applicabili, quindi scegli Salva. 11. Per aggiungere il nuovo utente di stampa al gruppo Amministratore, vai al pannello di navigazione, scegli Sicurezza, quindi scegli Configura. 12. Nella finestra di configura zione della sicurezza, aggiungi il tuo nuovo utente di stampa alla colonna Amministratore.	

Configura l'autenticazione e l'autorizzazione di stampa

Operazione	Description	Competenze richieste
Crea un dominio AWS Managed Microsoft AD con utenti e gruppi.	1. Crea una Active Directory su AWS Managed Microsoft AD seguendo le istruzion i da Crea la tua directory AWS Managed Microsoft AD nella documentazione di AWS Directory Service. 2. Distribuisci un' EC2 istanza (Active Directory manager) e installa gli strumenti di Active Directory per gestire AWS Managed Microsoft	Architetto del cloud

Operazione	Description	Competenze richieste
	AD seguendo le istruzioni della Fase 3: Distribuisci un' EC2 istanza per gestire AWS Managed Microsoft AD nella documentazione di AWS Directory Service. 3. **Note** Connettiti alla tua EC2 istanza seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. : Quando ti connetti all' EC2 istanza, inserisci le tue credenziali di amministratore (per la directory che hai creato nel primo passaggio) nella finestra Sicurezza di Windows. 4. Nel menu Start di Windows, in Strumenti di amministrazione di Windows, scegli Utenti e computer di Active Directory. 5. Crea un utente di stampa nel dominio Active Directory	

Operazione	Description	Competenze richieste
	seguendo i passaggi della documentazione relativa alla creazione di un utente nella documentazione del servizio AWS Directory.	
Unisciti a LRS VPSX/MFI EC2 a un dominio AWS Managed Microsoft AD.	Aggiungi LRS VPSX/MFI EC2 al tuo dominio AWS Managed Microsoft AD automatic amente (documentazione AWS Knowledge Center) o manualmente (document azione AWS Directory Service).	Architetto del cloud

Operazione	Description	Competenze richieste
Configura e integra LRS/DIS con AWS Managed Microsoft AD.	1. Connettiti alla tua VPSX/MFI EC2 istanza LRS seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. 2. Nel menu Start di Windows, apri l'interfaccia Web VPSX. 3. Nel riquadro di navigazione, scegli Sicurezza, quindi scegli Configura. 4. Nella pagina Configurazione della sicurezza, nella sezione Parametri di sicurezza, per Tipo di sicurezza, seleziona Interno. 5. Inserisci le tue preferenze per le altre opzioni nella sezione Parametri di sicurezza. 6. Aprite la cartella LRS Output Management dal menu Start di Microsoft Windows, scegliete Server Start, quindi scegliete Server Stop. 7. Accedete a LRS VPSX/MFI con il nome utente e la password di Active Directory.	Architetto del cloud

Prova un flusso di lavoro di stampa online

Operazione	Description	Competenze richieste
Avviate una richiesta di stampa online dall'app Micro Focus ACCT Demo.	1. **Note** Aprite l'emulatore di terminale TN327 0 nell'istanza di Micro Focus Enterpris e Server. EC2 (: Questo modello utilizza emulatori di terminale 3270.) 2. Connect all'emulatore di terminale TN327 0 (Rumba). Per l'indirizzo del nome host, usa 127.0.0.1 . Per Telnet Port, utilizzare 9270. 3. Dopo la connessione allo schermo 3270, premi CTRL +SHIFT+Z per cancellare lo schermo. 4. Per avviare l'applica zione ACCT Demo, nella schermata trasparente, inserisci ACCT. Si apre la schermata principale dell'applicazione ACCT Demo online (CICS). Nota: la schermata principale include opzioni di menu come Account file, To search by name, enter,	Architetto del cloud

Operazione	Description	Competenze richieste
	Request type, Account e Printer. 5. Per inviare una richiesta di stampa dall'applicazione ACCT Demo online (CICS), inserisci P nel campo del tipo di richiesta, 11111 nel campo dell'account e P275 nel campo della stampante. Assicurati di impostare il valore nel campo della stampante sul valore dell'ID del terminale della stampante CICS. 6. Premere Invio. Il messaggio «Richiesta di stampa pianificata» viene visualizzato nella parte inferiore dello schermo. Ciò conferma che una richiesta di stampa online è stata generata dall'applicazione ACCT Demo e inviata a LRS VPS/MFI per l'elaborazione della stampa.	

Operazione	Description	Competenze richieste
Controllate l'output di stampa in LRS VPSX/MFI.	1. Connettiti alla tua VPSX/MFI EC2 istanza LRS seguendo le istruzioni del Passaggio 2: Connettiti alla tua istanza nella EC2 documentazione di Amazon. 2. Nel menu Start di Windows, apri l'interfaccia Web VPSX. 3. Nel riquadro di navigazione, scegliete Stampanti, quindi scegliete Output Queue. Trova la coda di stampa P275 che hai creato in precedenza per la stampa online. 4. Per la coda di stampa (P275), nella colonna Spool ID, scegliete l'ID bobina per la richiesta nella coda della stampante. 5. Nella scheda Azioni, nella colonna COMANDO, scegliete Sfoglia. È ora possibile visualizzare l'output di stampa di un estratto conto con le colonne per Numero di conto, COGNOME, PRIMO, INDIRIZZO, TELEFONO, N. Carte emesse, data di emissione, importo e saldo.	Tecnico di test

Operazione	Description	Competenze richieste
	Per un esempio, vedi l'allegato online_print_output per questo modello.	

Risorse correlate

- Modernizzazione dell'output LRS (documentazione LRS)
- Concetti di rete VTAM (documentazione IBM)
- Riepilogo dei tipi di unità logiche (LU) (documentazione IBM)
- ANSI e controlli del trasporto delle macchine (documentazione IBM)
- Potenziamento dei carichi di lavoro mainframe aziendali su AWS con Micro Focus (blog AWS Partner Network)
- Crea una PAC Micro Focus Enterprise Server con Amazon EC2 Auto Scaling and Systems Manager (documentazione AWS Prescriptive Guidance)
- Flusso di dati AFP (Advanced Function Presentation) (documentazione IBM)
- Line Conditioned Data Stream (LCDS) (documentazione Compart)

Informazioni aggiuntive

Considerazioni

Durante il percorso di modernizzazione, è possibile prendere in considerazione un'ampia varietà di configurazioni per i processi mainframe online e l'output che generano. La piattaforma mainframe è stata personalizzata da ogni cliente e fornitore che la utilizza con requisiti particolari che influiscono direttamente sulla stampa. Ad esempio, la piattaforma attuale può incorporare IBM Advanced Function Presentation (AFP) o Xerox Line Condition Data Stream (LCDS) nel flusso di lavoro corrente. Inoltre, i caratteri di controllo del mainframe carriage e le parole di comando del canale possono influire sull'aspetto della pagina stampata e potrebbero richiedere una gestione speciale. Come parte del processo di pianificazione della modernizzazione, consigliamo di valutare e comprendere le configurazioni del proprio ambiente di stampa specifico.

Acquisizione dei dati di stampa

Questa sezione riassume i metodi di programmazione delle applicazioni CICS che è possibile utilizzare in un ambiente mainframe IBM per la stampa. VPSX/MFI I componenti LRS forniscono tecniche per consentire agli stessi programmi applicativi di creare dati nello stesso modo. La tabella seguente descrive come ogni metodo di programmazione dell'applicazione è supportato in un'applicazione CICS modernizzata in esecuzione in AWS e Micro Focus Enterprise Server con un server di stampa LRS VPSX/MFI .

Metodo	Description	Support per il metodo in un ambiente modernizzato
EXEC CICS INVIA TESTO.. o EXEC CICS SEND MAP..	Questi metodi CICS e VTAM sono responsabili della creazione e della fornitura di flussi di dati di stampa 3270/SCS a LUTYPE0 e dispositivi di stampa. LUTYPE1 LUTYPE3	Un'interfaccia API (Application Program Interface) online Print Exit (DFHUPRNT) di Micro Focus consente l'elaborazione dei dati di stampa entro il momento in cui i flussi di dati di stampa 3270/SCS vengono creati utilizzando uno di questi metodi. VPSX/MFI
EXEC CICS INVIA TESTO.. o EXEC CICS SEND MAP.. (con software mainframe IBM di terze parti)	I metodi CICS e VTAM sono responsabili della creazione e della fornitura di flussi di dati di stampa 3270/SCS a LUTYPE0 e dispositivi di stampa. LUTYPE1 LUTYPE3 I prodotti software di terze parti intercettano i dati di stampa, li convertono in dati di formato di stampa standard con un carattere di ASA/MCH controllo e inseriscono i dati sulla bobina JES per essere elaborati da sistemi di stampa basati su mainframe che utilizzano JES.	Un'API Micro Focus online Print Exit (DFHUPRNT) consente l'elaborazione dei dati di stampa entro il momento in VPSX/MFI cui i flussi di dati di stampa 3270/SCS vengono creati utilizzando uno di questi metodi.

EXEC CICS SPOOLOPEN	Questo metodo viene utilizzato dai programmi applicativi CICS per scrivere dati direttame nte nella bobina JES. I dati diventano quindi disponibili per essere elaborati da sistemi di stampa basati su mainframe che utilizzano JES.	Micro Focus Enterprise Server sposta i dati nello spool Enterprise Server dove possono essere elaborati da VPSX/MFI Batch Print Exit (LRSPRTE6) che sposta i dati su VPSX.
DRS/API	Un'interfaccia programmatica fornita da LRS viene utilizzata per scrivere dati di stampa su JES.	VPSX/MFI fornisce un'interf accia sostitutiva che trasferis ce i dati di stampa direttame nte su VPSX.

Controlli dello stato del parco stampanti

LRS VPSX/MFI (LRS LoadX) è in grado di eseguire controlli approfonditi dello stato di salute, tra cui la gestione dei dispositivi e l'ottimizzazione operativa. La gestione dei dispositivi è in grado di rilevare guasti in un dispositivo di stampa e indirizzare la richiesta di stampa a una stampante funzionante. Per ulteriori informazioni sui controlli approfonditi dello stato delle flotte di stampanti, consultate la documentazione LRS inclusa nella licenza del prodotto.

Autenticazione e autorizzazione alla stampa

LRS/DIS enables LRS applications to authenticate user IDs and passwords by using Microsoft Active Directory or an LDAP server. In addition to basic print authorization, LRS/DISpuò anche applicare controlli di sicurezza di stampa a livello granulare nei seguenti casi d'uso:

- Gestisci chi può sfogliare il lavoro della stampante.
- Gestisci il livello di navigazione dei lavori di altri utenti.
- Gestisci le attività operative. Ad esempio, sicurezza a livello di comando come hold/release, purge, edit, copy e reindirizzamento. La sicurezza può essere impostata dall'ID utente o dal gruppo (simile al gruppo AD o al gruppo LDAP).

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Sposta i file mainframe direttamente su Amazon S3 utilizzando Transfer Family

Luis Gustavo Dantas, Amazon Web Services

Riepilogo

Come parte del percorso di modernizzazione, puoi affrontare la sfida del trasferimento di file tra i tuoi server locali e il cloud Amazon Web Services (AWS). Il trasferimento di dati dai mainframe può essere una sfida importante perché i mainframe in genere non possono accedere a data store moderni come Amazon Simple Storage Service (Amazon S3), Amazon Elastic Block Store (Amazon EBS) o Amazon Elastic File System (Amazon EFS).

Molti clienti utilizzano risorse di staging intermedie, come server Linux, Unix o Windows locali, per trasferire file nel cloud AWS. Puoi evitare questo metodo indiretto utilizzando AWS Transfer Family con il Secure Shell (SSH) File Transfer Protocol (SFTP) per caricare i file mainframe direttamente su Amazon S3.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un cloud privato virtuale (VPC) con una sottorete raggiungibile dalla tua piattaforma legacy
- Un endpoint Transfer Family per il tuo VPC
- File VSAM (Mainframe Virtual Storage Access Method) convertiti in file sequenziali a lunghezza fissa (documentazione IBM)

Limitazioni

- Per impostazione predefinita, SFTP trasferisce i file in modalità binaria, il che significa che i file vengono caricati su Amazon S3 con la codifica EBCDIC preservata. Se il tuo file non contiene dati binari o compressi, puoi utilizzare il sottocomando sftp ascii (documentazione IBM) per convertire i file in testo durante il trasferimento.
- È necessario decomprimere i file mainframe (AWS Prescriptive Guidance) che contengono contenuti compressi e binari per utilizzare questi file nell'ambiente di destinazione.
- Le dimensioni degli oggetti Amazon S3 possono variare da un minimo di 0 byte a un massimo di 5 TB. Per ulteriori informazioni sulle funzionalità di Amazon S3, consulta Amazon S3. FAQs

Architecture

Stack tecnologico di origine

- Linguaggio Job control (JCL)
- Shell Unix z/OS e ISPF
- SFTP
- VSAM e file flat

Stack tecnologico Target

- Transfer Family
- Simple Storage Service (Amazon S3)
- Amazon Virtual Private Cloud (Amazon VPC) (Amazon VPC)

Architettura di destinazione

Il diagramma seguente mostra un'architettura di riferimento per l'utilizzo di Transfer Family con SFTP per caricare i file mainframe direttamente in un bucket S3.

Il diagramma mostra il flusso di lavoro seguente:

1. Utilizzi un job JCL per trasferire i file del mainframe dal mainframe legacy al cloud AWS tramite Direct Connect.
2. Direct Connect consente al traffico di rete di rimanere sulla rete globale AWS e di bypassare la rete Internet pubblica. Direct Connect migliora anche la velocità di rete, a partire da 50 Mbps e scalabile fino a 100 Gbps.
3. L'endpoint VPC consente le connessioni tra le risorse VPC e i servizi supportati senza utilizzare la rete Internet pubblica. L'accesso a Transfer Family e Amazon S3 raggiunge un'elevata disponibilità grazie alle interfacce di rete elastiche situate in due sottoreti private e zone di disponibilità.
4. Transfer Family autentica gli utenti e utilizza SFTP per ricevere i file dall'ambiente legacy e spostarli in un bucket S3.

Automazione e scalabilità

Una volta attivato il servizio Transfer Family, puoi trasferire un numero illimitato di file dal mainframe ad Amazon S3 utilizzando un job JCL come client SFTP. Puoi anche automatizzare il trasferimento dei file utilizzando un programma di pianificazione dei processi in batch del mainframe per eseguire i processi SFTP quando sei pronto a trasferire i file mainframe.

Tools (Strumenti)

- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito. Questa rete virtuale è simile a una rete tradizionale che gestiresti nel tuo data center, con i vantaggi dell'utilizzo dell'infrastruttura scalabile di AWS.
- AWS Transfer Family ti consente di scalare in modo sicuro i trasferimenti ricorrenti di business-to-business file verso Amazon S3 e Amazon EFS utilizzando i protocolli SFTP, FTPS e FTP.

Epiche

Crea il bucket S3 e la politica di accesso

Operazione	Description	Competenze richieste
Crea il bucket S3.	Crea un bucket S3 per ospitare i file che trasferisci dal tuo ambiente legacy.	Informazioni generali su AWS
Crea il ruolo e la policy di IAM.	Transfer Family utilizza il tuo ruolo AWS Identity and Access Management (IAM) per concedere l'accesso al bucket S3 che hai creato in precedenza. Crea un ruolo IAM che includa la seguente policy IAM: `{ "Version": "2012-10-17",`	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	"Statement": [{ "Sid": "UserFolderListing", "Action": ["s3:ListBucket", "s3:GetBucketLocation"], "Effect": "Allow", "Resource": ["arn:aws:s3:::<your-bucket-name>"] }, { "Sid": "HomeDirObjectAccess", "Effect": "Allow", "Action": ["s3:PutObject", "s3:GetObjectAcl", "s3:GetObject", "s3:DeleteObjectVersion", "s3:DeleteObject", "s3:PutObjectAcl", "s3:GetObjectVersion"	

Operazione	Description	Competenze richieste

	<pre>],
	"Resource":
	"arn:aws:s3:::<your-bucket-name>/*"
	}
	]
	}</pre>
	Note
	Devi scegliere lo use case Transfer quando crei il ruolo IAM.	

Definisci il servizio di trasferimento

Operazione	Description	Competenze richieste
Crea il server SFTP.	1. Accedi alla console di gestione AWS, apri la console Transfer Family, quindi scegli Create server. 2. Scegli solo SFTP (SSH File Transfer Protocol) - trasferimento di file tramite protocollo Secure Shell, quindi scegli Avanti. 3. Per Identity provider, scegli Servizio gestito, quindi scegli Avanti. 4. Per il tipo di endpoint, scegli VPC ospitato. 5. Per Access, scegli Interno. 6. In VPC, seleziona il VPC.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	7. Nella sezione Zone di disponibilità, scegli le zone di disponibilità e le sottoreti. 8. Nella sezione Gruppi di sicurezza, scegli il tuo gruppo di sicurezza, quindi scegli Avanti. 9. Per Dominio, scegli Amazon S3, quindi scegli Avanti. 10. Lascia le opzioni predefinite nella pagina Configura dettagli aggiuntivi, quindi scegli Avanti. 11. Scegliere Create Server (Crea server). **Note** Per ulteriori informazioni su come configurare un server SFTP, consulta Creare un server compatibile con SFTP (AWS Transfer Family User Guide).	

Operazione	Description	Competenze richieste
Ottieni l'indirizzo del server.	1. Apri la console Transfer Family e scegli l'ID del tuo server nella colonna Server ID. 2. Nella sezione Dettagli dell'endpoint, per Tipo di endpoint, scegli l'ID dell'endpoint. Questo ti porta alla console Amazon VPC. 3. Nella scheda Dettagli della console Amazon VPC, trova i nomi DNS accanto ai nomi DNS.	Informazioni generali su AWS
Crea la key pair del client SFTP.	Crea una coppia di chiavi SSH per Microsoft Windows o macOS/Linux/UNIX.	AWS generale, SSH

Operazione	Description	Competenze richieste
Crea l'utente SFTP.	1. Apri la console Transfer Family, scegli Server dal pannello di navigazione, quindi seleziona il tuo server. 2. Nella colonna Server ID, scegli l'ID server per il tuo server, quindi scegli Aggiungi utente. 3. Per Username, inserisci un nome utente che corrispon da al nome utente della tua coppia di key pair SSH. 4. Per Ruolo, scegli il ruolo IAM che hai creato in precedenza. 5. Per Home directory, scegli il bucket S3 che hai creato in precedenza. 6. Per le chiavi pubbliche SSH, inserisci la coppia di chiavi che hai creato in precedenza. 7. Scegliere Aggiungi.	Informazioni generali su AWS

Trasferisci il file mainframe

Operazione	Description	Competenze richieste
Invia la chiave privata SSH al mainframe.	Usa SFTP o SCP per inviare la chiave privata SSH all'ambiente legacy.	Mainframe, shell z/OS Unix, FTP, SCP

Operazione	Description	Competenze richieste

	Esempio SFTP:
	`sftp [USERNAME@mainframeIP]`
	`[password]`
	`cd [/u/USERNAME]`
	`put [your-key-pair-file]`
	Esempio SCP:
	`scp [your-key-pair-file] [USERNAME@MainframeIP]:/[u/USERNAME]`
	Quindi, memorizza la chiave SSH nel file system z/OS Unix con il nome utente che successivamente eseguirà il processo batch di trasferimento dei file (ad esempio,). `/u/CONTROLM`
	Note
	Per ulteriori informazioni sulla shell z/OS Unix, vedere Introduzione alle z/OS shell (documentazione IBM).	

Operazione	Description	Competenze richieste

Crea il client SFTP JCL.	Poiché i mainframe non dispongono di un client SFTP nativo, è necessario utilizzar e l'utilità BPXBATCH per eseguire il client SFTP dalla shell Unix. z/OS
	
	Nell'editor ISPF, create il client SFTP JCL. Esempio:
	
	`//JOBNAM JOB ...`
	`//************`
	`******************`
	`******************`
	`******************`
	`****`
	`//SFTP EXEC PGM=BPXBA`
	`TCH,REGION=0M`
	`//STDPARM DD *`
	`SH cp "//'MAINF`
	`RAME.FILE.NAME'"`
	` filename.txt;`
	`echo 'put filename.txt'`
	` > uplcmd;`
	`sftp -b uplcmd -i`
	` ssh_private_key_fi`
	`le ssh_username@<tran`
	`sfer service ip or`
	` DNS>;`
	`//SYSPRINT DD SYSOUT=*`
	`//STDOUT DD SYSOUT=*`
	`//STDENV DD *`
	`//STDERR DD SYSOUT=*`
	
	Note
	Per ulteriori informazi oni su come eseguire un comando nella
		JCL, mainframe, shell Unix z/ OS

Operazione	Description	Competenze richieste
	shell z/OS Unix, vedere L'utilità BPXBATCH (documentazione IBM). Per ulteriori informazioni su come creare o modificare i job JCL in z/OS, consulta Cos'è ISPF? e L'editor ISPF (documentazione IBM).	
Esegui il client SFTP JCL.	1. Nell'editor ISPF, immettete SUB, quindi premete il tasto ENTER dopo la creazione del lavoro JCL. 2. Monitora l'attività del processo batch di trasferimento dei file del mainframe in SDSF. **Note** Per ulteriori informazioni su come controllare l'attività dei processi in batch, consulta la Guida per l'utente di z/OS SDSF (documentazione IBM).	Mainframe, JCL, ISPF

Operazione	Description	Competenze richieste
Convalida il trasferimento dei file.	1. Accedi alla console di gestione AWS, apri la console Amazon S3, quindi scegli Bucket dal pannello di navigazione. 2. Scegli il bucket associato al tuo Transfer Family. 3. Nella sezione Oggetti della scheda Oggetti, trova il file che hai trasferito dal mainframe.	Informazioni generali su AWS
Automatizza il client SFTP JCL.	Usa Job Scheduler per attivare automaticamente il client SFTP JCL. **Note** È possibile utilizzare strumenti di pianificazione dei processi mainframe, come BMC Control-M o CA Workload Automation, per automatizzare i processi in batch per i trasferimenti di file in base all'ora e ad altre dipendenze dei processi batch.	Job scheduler

Risorse correlate

- Come funziona AWS Transfer Family

Ottimizza le prestazioni della tua applicazione modernizzata AWS Blu Age

Vishal Jaswani, Manish Roy e Himanshu Sah, Amazon Web Services

Riepilogo

Le applicazioni mainframe modernizzate con AWS Blu Age richiedono test di equivalenza funzionale e prestazionale prima di essere implementate in produzione. Nei test delle prestazioni, le applicazioni modernizzate possono funzionare più lentamente rispetto ai sistemi legacy, in particolare nei lavori in batch complessi. Questa disparità esiste perché le applicazioni mainframe sono monolitiche, mentre le applicazioni moderne utilizzano architetture a più livelli. Questo modello presenta tecniche di ottimizzazione per colmare queste lacune prestazionali per le applicazioni modernizzate utilizzando il refactoring automatizzato con Blu Age. AWS

Il modello utilizza il framework di modernizzazione AWS Blu Age con Java nativo e funzionalità di ottimizzazione del database per identificare e risolvere i problemi di prestazioni. Il modello descrive come utilizzare la profilazione e il monitoraggio per identificare problemi di prestazioni con metriche quali i tempi di esecuzione SQL, l'utilizzo della memoria e i modelli. I/O Spiega quindi come applicare ottimizzazioni mirate, tra cui la ristrutturazione delle query del database, la memorizzazione nella cache e il perfezionamento della logica aziendale.

I miglioramenti nei tempi di elaborazione in batch e nell'utilizzo delle risorse di sistema aiutano a raggiungere i livelli di prestazioni del mainframe nei sistemi modernizzati. Questo approccio mantiene l'equivalenza funzionale durante la transizione verso le moderne architetture basate sul cloud.

Per utilizzare questo modello, configura il sistema e identifica gli hotspot prestazionali seguendo le istruzioni nella sezione Epics e applica le tecniche di ottimizzazione descritte in dettaglio nella sezione Architettura.

Prerequisiti e limitazioni

Prerequisiti

- Un'applicazione AWS modernizzata di Blu Age
- Una licenza JProfiler
- Privilegi amministrativi per installare client di database e strumenti di profilazione
- AWS Certificazione Blu Age Level 3
- Comprensione di livello intermedio del framework AWS Blu Age, della struttura del codice generato e della programmazione Java

Limitazioni

Le seguenti funzionalità e funzionalità di ottimizzazione non rientrano nell'ambito di questo modello:

- Ottimizzazione della latenza di rete tra i livelli di applicazione
- Ottimizzazioni a livello di infrastruttura tramite tipi di istanze Amazon Elastic Compute Cloud (Amazon EC2) e ottimizzazione dello storage
- Test di carico utente e test di stress simultanei

Versioni del prodotto

- JProfiler versione 13.0 o successiva (consigliamo la versione più recente)
- pgAdmin versione 8.14 o successiva

Architecture

Questo modello imposta un ambiente di profilazione per un'applicazione AWS Blu Age utilizzando strumenti come JProfiler pgAdmin. Supporta l'ottimizzazione tramite DAOManager and SQLExecution Builder APIs fornito da Blu Age. AWS

La parte restante di questa sezione fornisce informazioni dettagliate ed esempi per identificare gli hotspot prestazionali e le strategie di ottimizzazione per le applicazioni modernizzate. I passaggi della sezione Epics fanno riferimento a queste informazioni per ulteriori indicazioni.

Identificazione degli hotspot prestazionali nelle applicazioni mainframe modernizzate

Nelle applicazioni mainframe modernizzate, gli hotspot prestazionali sono aree specifiche del codice che causano rallentamenti o inefficienze significativi. Questi hotspot sono spesso causati dalle differenze architettoniche tra le applicazioni mainframe e quelle modernizzate. Per identificare questi ostacoli prestazionali e ottimizzare le prestazioni dell'applicazione modernizzata, è possibile utilizzare tre tecniche: registrazione SQL, piano di query e analisi. `EXPLAIN` JProfiler

Tecnica di identificazione degli hotspot: registrazione SQL

Le moderne applicazioni Java, comprese quelle che sono state modernizzate utilizzando AWS Blu Age, dispongono di funzionalità integrate per registrare le query SQL. È possibile abilitare logger specifici nei progetti AWS Blu Age per tracciare e analizzare le istruzioni SQL eseguite dall'applicazione. Questa tecnica è particolarmente utile per identificare modelli di accesso al

database inefficienti, come query individuali eccessive o chiamate al database mal strutturate, che potrebbero essere ottimizzati mediante il raggruppamento in batch o il perfezionamento delle query.

Per implementare la registrazione SQL nella tua applicazione modernizzata AWS Blu Age, imposta il livello di registro su affinché le istruzioni SQL nel file acquisiscano `DEBUG` i dettagli sull'esecuzione delle `application.properties` query:

```
level.org.springframework.beans.factory.support.DefaultListableBeanFactory : WARN
level.com.netfective.bluage.gapwalk.runtime.sort.internal: WARN
level.org.springframework.jdbc.core.StatementCreatorUtils: DEBUG
level.com.netfective.bluage.gapwalk.rt.blu4iv.dao: DEBUG
level.com.fiserv.signature: DEBUG
level.com.netfective.bluage.gapwalk.database.support.central: DEBUG
level.com.netfective.bluage.gapwalk.rt.db.configuration.DatabaseConfiguration: DEBUG
level.com.netfective.bluage.gapwalk.rt.db.DatabaseInteractionLoggerUtils: DEBUG
level.com.netfective.bluage.gapwalk.database.support.AbstractDatabaseSupport: DEBUG
level.com.netfective.bluage.gapwalk.rt: DEBUG
```

Monitora le query ad alta frequenza e con prestazioni lente utilizzando i dati registrati per identificare gli obiettivi di ottimizzazione. Concentrati sulle query all'interno dei processi batch perché in genere hanno il massimo impatto sulle prestazioni.

Tecnica di identificazione degli hotspot: piano Query EXPLAIN

Questo metodo utilizza le funzionalità di pianificazione delle interrogazioni dei sistemi di gestione di database relazionali. È possibile utilizzare comandi come `EXPLAIN` PostgreSQL o MySQL `EXPLAIN PLAN` o Oracle per esaminare in che modo il database intende eseguire una determinata query. L'output di questi comandi fornisce informazioni preziose sulla strategia di esecuzione delle query, incluso se verranno utilizzati gli indici o verranno eseguite scansioni complete della tabella. Queste informazioni sono fondamentali per ottimizzare le prestazioni delle query, specialmente nei casi in cui un'indicizzazione corretta può ridurre significativamente i tempi di esecuzione.

Estrai le query SQL più ripetitive dai log delle applicazioni e analizza il percorso di esecuzione delle query con prestazioni lente utilizzando il comando specifico del tuo database. `EXPLAIN` Ecco un esempio di database PostgreSQL.

Query:

```
SELECT * FROM tenk1 WHERE unique1 < 100;
```

`EXPLAIN`comando:

```
EXPLAIN SELECT * FROM tenk1 where unique1 < 100;
```

Output:

```
Bitmap Heap Scan on tenk1 (cost=5.06..224.98 rows=100 width=244)
Recheck Cond: (unique1 < 100)
-> Bitmap Index Scan on tenk1_unique1 (cost=0.00..5.04 rows=100 width=0)
Index Cond: (unique1 < 100)
```

È possibile interpretare l'EXPLAINoutput come segue:

- Leggete il EXPLAIN piano dalle operazioni più interne a quelle più esterne (dal basso verso l'alto).
- Cerca i termini chiave. Ad esempio, `Seq Scan` indica la scansione completa della tabella e `Index Scan` mostra l'utilizzo dell'indice.
- Verifica i valori dei costi: il primo numero è il costo iniziale e il secondo numero è il costo totale.
- Vedi il `rows` valore per il numero stimato di righe di output.

In questo esempio, il motore di query utilizza una scansione dell'indice per trovare le righe corrispondenti, quindi recupera solo quelle righe (`Bitmap Heap Scan`). Ciò è più efficiente rispetto alla scansione dell'intera tabella, nonostante il costo più elevato dell'accesso alle singole righe.

Le operazioni di scansione delle tabelle nell'output di un EXPLAIN piano indicano un indice mancante. L'ottimizzazione richiede la creazione di un indice appropriato.

Tecnica di identificazione degli hotspot: analisi JProfiler

JProfiler è uno strumento completo di profilazione Java che consente di risolvere i problemi di prestazioni identificando le chiamate lente al database e le chiamate che richiedono un uso intensivo della CPU. Questo strumento è particolarmente efficace per identificare le query SQL lente e l'utilizzo inefficiente della memoria.

Esempio di analisi per le interrogazioni:

```
select evt.
 com.netfective.bluage.gapwalk.rt.blu4iv.dao.Blu4ivTableManager.queryNonTrasactional
```

La visualizzazione JProfiler Hot Spots fornisce le seguenti informazioni:

- Colonna relativa all'ora
  - Mostra la durata totale dell'esecuzione (ad esempio, 329 secondi)
  - Visualizza la percentuale del tempo totale di applicazione (ad esempio, 58,7%)
  - Aiuta a identificare le operazioni che richiedono più tempo
- Colonna Tempo medio
  - Mostra la durata per esecuzione (ad esempio, 2.692 microsecondi)
  - Indica le prestazioni delle singole operazioni
  - Aiuta a individuare le singole operazioni lente
- Colonna Eventi
  - Mostra il conteggio delle esecuzioni (ad esempio, 122.387 volte)
  - Indica la frequenza operativa
  - Aiuta a identificare i metodi chiamati di frequente

Per i risultati di esempio:

- Alta frequenza: 122.387 esecuzioni indicano un potenziale di ottimizzazione
- Problema relativo alle prestazioni: il tempo medio di 2.692 microsecondi indica un'inefficienza
- Impatto critico: il 58,7% del tempo totale indica un grave ostacolo

JProfiler è in grado di analizzare il comportamento di esecuzione dell'applicazione per individuare punti critici che potrebbero non essere visibili mediante l'analisi statica del codice o la registrazione SQL. Queste metriche aiutano a identificare le operazioni che necessitano di ottimizzazione e a determinare la strategia di ottimizzazione più efficace. Per ulteriori informazioni sulle JProfiler funzionalità, consulta la JProfiler documentazione.

Quando si utilizzano queste tre tecniche (registrazione SQL, `EXPLAIN` piano di query e JProfiler) in combinazione, è possibile ottenere una visione olistica delle caratteristiche prestazionali dell'applicazione. Identificando e risolvendo gli hotspot prestazionali più critici, è possibile colmare il divario prestazionale tra l'applicazione mainframe originale e il sistema modernizzato basato sul cloud.

Dopo aver identificato gli hotspot prestazionali dell'applicazione, è possibile applicare le strategie di ottimizzazione, illustrate nella sezione successiva.

Strategie di ottimizzazione per la modernizzazione del mainframe

Questa sezione descrive le strategie chiave per ottimizzare le applicazioni che sono state modernizzate dai sistemi mainframe. Si concentra su tre strategie: utilizzo delle strategie esistenti APIs, implementazione di un caching efficace e ottimizzazione della logica aziendale.

Strategia di ottimizzazione: utilizzo delle soluzioni esistenti APIs

AWS Blu Age offre diverse potenti APIs interfacce DAO che è possibile utilizzare per ottimizzare le prestazioni. Due interfacce principali, DAOManager e SQLExecution Builder, offrono funzionalità per migliorare le prestazioni delle applicazioni.

DAOManager

DAOManager funge da interfaccia principale per le operazioni di database nelle applicazioni modernizzate. Offre diversi metodi per migliorare le operazioni del database e migliorare le prestazioni delle applicazioni, in particolare per le semplici operazioni di creazione, lettura, aggiornamento ed eliminazione (CRUD) e l'elaborazione in batch.

- Usa. SetMaxResults Nell' DAOManager API, è possibile utilizzare il SetMaxResultsmetodo per specificare il numero massimo di record da recuperare in una singola operazione di database. Per impostazione predefinita, DAOManager recupera solo 10 record alla volta, il che può portare a più chiamate al database durante l'elaborazione di set di dati di grandi dimensioni. Utilizzate questa ottimizzazione quando l'applicazione deve elaborare un gran numero di record e attualmente sta effettuando più chiamate al database per recuperarli. Ciò è particolarmente utile negli scenari di elaborazione in batch in cui si esegue l'iterazione su un set di dati di grandi dimensioni. Nell'esempio seguente, il codice a sinistra (prima dell'ottimizzazione) utilizza il valore di recupero dei dati predefinito di 10 record. Il codice a destra (dopo l'ottimizzazione) è impostato setMaxResultsper recuperare 100.000 record alla volta.

> **Note**
>
> Scegliete con attenzione batch di dimensioni maggiori e controllate le dimensioni degli oggetti, poiché questa ottimizzazione aumenta l'ingombro di memoria.

- Sostituisci SetOnGreatorOrEqual con SetOnEqual. Questa ottimizzazione comporta la modifica del metodo utilizzato per impostare la condizione per il recupero dei record. Il SetOnGreatorOrEqualmetodo recupera i record che sono maggiori o uguali a un valore specificato, mentre SetOnEqualrecupera solo i record che corrispondono esattamente al valore specificato.

Utilizza SetOnEqualcome illustrato nel seguente esempio di codice, quando sai di aver bisogno di corrispondenze esatte e stai attualmente utilizzando il SetOnGreatorOrEqualmetodo seguito da readNextEqual (). Questa ottimizzazione riduce il recupero di dati non necessario.

- Utilizza operazioni di scrittura e aggiornamento in batch. È possibile utilizzare le operazioni batch per raggruppare più operazioni di scrittura o aggiornamento in un'unica transazione di database. Ciò riduce il numero di chiamate al database e può migliorare significativamente le prestazioni per le operazioni che coinvolgono più record.

  Nell'esempio seguente, il codice a sinistra esegue operazioni di scrittura in un ciclo, il che rallenta le prestazioni dell'applicazione. È possibile ottimizzare questo codice utilizzando un'operazione di scrittura in batch: durante ogni iterazione del `WHILE` ciclo, si aggiungono record a un batch finché la dimensione del batch non raggiunge una dimensione predeterminata di 100. È quindi possibile svuotare il batch quando raggiunge la dimensione predeterminata e quindi scaricare tutti i record rimanenti nel database. Ciò è particolarmente utile negli scenari in cui si elaborano set di dati di grandi dimensioni che richiedono aggiornamenti.

- Aggiungere indici. L'aggiunta di indici è un'ottimizzazione a livello di database che può migliorare significativamente le prestazioni delle query. Un indice consente al database di individuare rapidamente le righe con un valore di colonna specifico senza eseguire la scansione dell'intera tabella. Utilizza l'indicizzazione sulle colonne utilizzate di frequente in `WHERE` clausole, `JOIN` condizioni o istruzioni. `ORDER BY` Ciò è particolarmente importante per tabelle di grandi dimensioni o quando il recupero rapido dei dati è fondamentale.

SQLExecutionCostruttore

SQLExecutionBuilder è un'API flessibile che puoi utilizzare per assumere il controllo delle query SQL che verranno eseguite, recuperare solo determinate colonne, utilizzare e utilizzare nomi `INSERT` di tabelle `SELECT` dinamici. Nell'esempio seguente, SQLExecutor Builder utilizza una query personalizzata definita dall'utente.

Scelta tra DAOManager e Builder SQLExecution

La scelta tra questi APIs dipende dal caso d'uso specifico:

- Usalo DAOManager quando vuoi che AWS Blu Age Runtime generi le query SQL invece di scriverle tu stesso.
- Scegli SQLExecution Builder quando devi scrivere query SQL per sfruttare le funzionalità specifiche del database o scrivere query SQL ottimali.

Strategia di ottimizzazione: caching

Nelle applicazioni modernizzate, l'implementazione di strategie di caching efficaci può ridurre significativamente le chiamate al database e migliorare i tempi di risposta. Questo aiuta a colmare il divario di prestazioni tra ambienti mainframe e cloud.

Nelle applicazioni AWS Blu Age, le semplici implementazioni di caching utilizzano strutture di dati interne come mappe hash o elenchi di array, quindi non è necessario configurare una soluzione di caching esterna che richieda una ristrutturazione dei costi e del codice. Questo approccio è particolarmente efficace per i dati a cui si accede frequentemente ma che vengono modificati raramente. Quando implementi la memorizzazione nella cache, considera i vincoli di memoria e i modelli di aggiornamento per garantire che i dati memorizzati nella cache rimangano coerenti e offrano vantaggi effettivi in termini di prestazioni.

La chiave per una corretta memorizzazione nella cache è identificare i dati giusti da memorizzare nella cache. Nell'esempio seguente, il codice a sinistra legge sempre i dati dalla tabella, mentre il codice a destra legge i dati dalla tabella quando la mappa hash locale non ha un valore per una determinata chiave. `cacheMap`è un oggetto di mappa hash creato nel contesto del programma e cancellato nel metodo di pulizia del contesto del programma.

Memorizzazione nella cache con: DAOManager

Memorizzazione nella cache con Builder: SQLExecution

Strategia di ottimizzazione: ottimizzazione della logica aziendale

L'ottimizzazione della logica aziendale si concentra sulla ristrutturazione del codice generato automaticamente da AWS Blu Age per allinearlo meglio alle funzionalità dell'architettura moderna. Ciò diventa necessario quando il codice generato mantiene la stessa struttura logica del codice mainframe legacy, il che potrebbe non essere ottimale per i sistemi moderni. L'obiettivo è migliorare le prestazioni mantenendo l'equivalenza funzionale con l'applicazione originale.

Questo approccio di ottimizzazione va oltre le semplici modifiche alle API e le strategie di memorizzazione nella cache. Implica modifiche al modo in cui l'applicazione elabora i dati e interagisce con il database. Le ottimizzazioni più comuni includono l'eliminazione di operazioni di lettura non necessarie per semplici aggiornamenti, la rimozione delle chiamate ridondanti al database e la ristrutturazione dei modelli di accesso ai dati per allinearli meglio alla moderna architettura applicativa. Di seguito si riportano alcuni esempi:

- Aggiornamento dei dati direttamente nel database.Ristruttura la logica di business utilizzando gli aggiornamenti SQL diretti anziché più DAOManager operazioni con loop. Ad esempio, il codice seguente (lato sinistro) effettua più chiamate al database e utilizza una quantità eccessiva di memoria. In particolare, utilizza più operazioni di lettura e scrittura del database all'interno dei loop, aggiornamenti individuali anziché l'elaborazione in batch e la creazione di oggetti non necessari per ogni iterazione.

  Il seguente codice ottimizzato (lato destro) utilizza una singola operazione di aggiornamento Direct SQL. In particolare, utilizza una singola chiamata al database anziché più chiamate e non richiede cicli perché tutti gli aggiornamenti vengono gestiti in un'unica istruzione. Questa ottimizzazione offre prestazioni e utilizzo delle risorse migliori e riduce la complessità. Previene l'iniezione di SQL, offre una migliore memorizzazione nella cache del piano di query e aiuta a migliorare la sicurezza.

  > **Note**
  >
  > Utilizza sempre query parametrizzate per impedire l'iniezione di SQL e garantire una corretta gestione delle transazioni.

- Riduzione delle chiamate ridondanti al database. Le chiamate ridondanti al database possono influire in modo significativo sulle prestazioni delle applicazioni, in particolare quando avvengono all'interno di loop. Una tecnica di ottimizzazione semplice ma efficace consiste nell'evitare di ripetere più volte la stessa query sul database. Il seguente confronto di codice dimostra come lo spostamento della chiamata al `retrieve()` database all'esterno del ciclo impedisca l'esecuzione ridondante di query identiche, migliorando l'efficienza.

- Riduzione delle chiamate al database utilizzando la clausola SQL. **`JOIN`** Implementa SQLExecution Builder per ridurre al minimo le chiamate al database. SQLExecutionBuilder offre un maggiore controllo sulla generazione di SQL ed è particolarmente utile per query complesse

che DAOManager non possono essere gestite in modo efficiente. Ad esempio, il codice seguente utilizza più DAOManager chiamate:

```
List<Employee> employees = daoManager.readAll();
for(Employee emp : employees) {
    Department dept = deptManager.readById(emp.getDeptId());  // Additional call for
 each employee
    Project proj = projManager.readById(emp.getProjId());     // Another call for
 each employee
    processEmployeeData(emp, dept, proj);
}
```

Il codice ottimizzato utilizza una singola chiamata al database in SQLExecution Builder:

```
SQLExecutionBuilder builder = new SQLExecutionBuilder();
builder.append("SELECT e.*, d.name as dept_name, p.name as proj_name");
builder.append("FROM employee e");
builder.append("JOIN department d ON e.dept_id = d.id");
builder.append("JOIN project p ON e.proj_id = p.id");
builder.append("WHERE e.status = ?", "ACTIVE");

List<Map<String, Object>> results = builder.execute();  // Single database call
for(Map<String, Object> result : results) {
    processComplexData(result);
}
```

Utilizzo congiunto di strategie di ottimizzazione

Queste tre strategie funzionano in modo sinergico: APIs forniscono gli strumenti per un accesso efficiente ai dati, la memorizzazione nella cache riduce la necessità di recuperi ripetuti dei dati e l'ottimizzazione della logica aziendale garantisce che questi APIs vengano utilizzati nel modo più efficace possibile. Il monitoraggio e la regolazione regolari di queste ottimizzazioni garantiscono continui miglioramenti delle prestazioni, pur mantenendo l'affidabilità e la funzionalità dell'applicazione modernizzata. La chiave del successo sta nel capire quando e come applicare ciascuna strategia in base alle caratteristiche e agli obiettivi prestazionali dell'applicazione.

Tools (Strumenti)

- JProfilerè uno strumento di profilazione Java progettato per sviluppatori e ingegneri delle prestazioni. Analizza le applicazioni Java e aiuta a identificare rallentamenti nelle prestazioni,

perdite di memoria e problemi di threading. JProfiler offre la profilazione di CPU, memoria e thread, nonché il monitoraggio di database e macchine virtuali Java (JVM) per fornire informazioni sul comportamento delle applicazioni.

> **Note**
>
> In alternativa JProfiler, è possibile utilizzare Java VisualVM. Si tratta di uno strumento gratuito e open source di profilazione e monitoraggio delle prestazioni per applicazioni Java che offre il monitoraggio in tempo reale dell'utilizzo della CPU, del consumo di memoria, della gestione dei thread e delle statistiche sulla raccolta dei rifiuti. Poiché Java VisualVM è uno strumento JDK integrato, è più conveniente rispetto alle esigenze di profilazione di base. JProfiler

- pgAdmin è uno strumento di amministrazione e sviluppo open source per PostgreSQL. Fornisce un'interfaccia grafica che consente di creare, gestire e utilizzare oggetti di database. È possibile utilizzare pgAdmin per eseguire un'ampia gamma di attività, dalla scrittura di semplici query SQL allo sviluppo di database complessi. Le sue funzionalità includono un editor SQL che evidenzia la sintassi, un editor di codice lato server, un agente di pianificazione per attività SQL, shell e batch e il supporto per tutte le funzionalità di PostgreSQL sia per utenti PostgreSQL principianti che esperti.

Best practice

Identificazione degli hotspot prestazionali:

- Documenta le metriche prestazionali di base prima di iniziare le ottimizzazioni.
- Stabilisci obiettivi chiari di miglioramento delle prestazioni in base ai requisiti aziendali.
- Durante il benchmarking, disabilita la registrazione dettagliata, poiché può influire sulle prestazioni.
- Configura una suite di test delle prestazioni ed eseguila periodicamente.
- Usa l'ultima versione di pgAdmin. (Le versioni precedenti non supportano il piano di `EXPLAIN` interrogazione).
- Per il benchmarking, scollegatelo JProfiler dopo aver completato le ottimizzazioni perché aumenta la latenza.
- Per il benchmarking, assicuratevi di eseguire il server in modalità start anziché in modalità debug, perché la modalità di debug aumenta la latenza.

Strategie di ottimizzazione:

- Configura SetMaxResultsi valori nel `application.yaml` file per specificare i batch delle dimensioni corrette in base alle specifiche del sistema.
- Configura SetMaxResultsi valori in base al volume di dati e ai vincoli di memoria.
- Passa SetOnGreatorOrEquala SetOnEqualsolo in caso di chiamate successive. `.readNextEqual()`
- Nelle operazioni di scrittura o aggiornamento in batch, gestite l'ultimo batch separatamente, poiché potrebbe essere inferiore alla dimensione del batch configurato e potrebbe non essere rilevato dall'operazione di scrittura o aggiornamento.

Memorizzazione nella cache:

- I campi introdotti per la memorizzazione nella cache`processImpl`, che mutano ad ogni esecuzione, devono sempre essere definiti nel contesto di tale operazione. `processImpl` I campi devono essere cancellati anche utilizzando il `doReset()` metodo or. `cleanUp()`
- Quando implementate la memorizzazione nella cache in memoria, dimensionate correttamente la cache. Le cache molto grandi archiviate in memoria possono occupare tutte le risorse, il che potrebbe influire sulle prestazioni complessive dell'applicazione.

SQLExecutionCostruttore:

- Per le query che intendi utilizzare in SQLExecution Builder, utilizza nomi chiave come. `PROGRAMNAME_STATEMENTNUMBER`
- Quando usi SQLExecution Builder, controlla sempre il campo. `Sqlcod` Questo campo contiene un valore che specifica se la query è stata eseguita correttamente o se sono stati rilevati errori.
- Utilizza query con parametri per impedire l'iniezione di SQL.

Ottimizzazione della logica aziendale:

- Mantieni l'equivalenza funzionale durante la ristrutturazione del codice ed esegui test di regressione e confronto tra database per il sottoinsieme di programmi pertinente.
- Conserva istantanee di profilazione per il confronto.

Epiche

Installazione JProfiler e pgAdmin

Operazione	Description	Competenze richieste
Installa e configura JProfiler.	1. Nella pagina di JProfiler download, scarica il file JProfiler per il tuo sistema operativo (Windows, macOS o Linux). 2. Completa il processo di installazione seguendo le istruzioni contenute nella JProfiler documentazione. 3. Esegui la configurazione iniziale: a. Avvia l' JProfiler applicazione. b. Attiva la chiave di licenza. c. Configura le impostazioni JVM predefinite. 4. Verifica l'installazione: a. Avvio. JProfiler b. Esegui una sessione di test di profilazione. Per istruzioni, consulta Collegamento ai servizi locali nella sezione Profilazione di una JVM della documentazione. JProfiler	Sviluppatore di app
Installa e configura pgAdmin.	In questo passaggio, si installa e si configura un client DB	Sviluppatore di app

Operazione	Description	Competenze richieste
	per interrogare il database. Questo modello utilizza un database PostgreSQL e pgAdmin come client di database. Se utilizzi un altro motore di database, segui la documentazione per il client DB corrispondente. 1. Scarica e installa pgAdmin per il tuo sistema operativo dalla pagina di download di pgAdmin. Imposta la password principale durante l'installazione. 2. Connect al server del database. Per istruzioni, consulta la documentazione di pgAdmin. 3. Verifica l'installazione utilizzando pGAdmin Query Tool per eseguire query SQL di base.	

Identifica gli hotspot

Operazione	Description	Competenze richieste
Abilita la registrazione delle query SQL nell'applicazione AWS Blu Age.	Abilita i logger per la registrazione delle query SQL nel `application.properties` file dell'applicazione AWS Blu Age, come spiegato nella sezione Architettura.	Sviluppatore di app

Operazione	Description	Competenze richieste
Genera e analizza i `EXPLAIN` piani di interrogazione per identificare gli hotspot prestazionali del database.	Per i dettagli, consulta la sezione Architettura.	Sviluppatore di app

Operazione	Description	Competenze richieste
Crea un' JProfiler istantanea per analizzare un test case con prestazioni lente.	1. Avvia: JProfiler a. Apri l' JProfiler applicazione. b. In Start Center, scegli la scheda Quick Attach. c. Seleziona il processo Tomcat dall'elenco. d. Scegli Avvia. 2. Configura le impostazioni di profilazione: a. Scegli l'opzione di campionamento consigliata. b. Mantieni le opzioni predefinite nella schermata di configurazione, quindi scegli OK. c. Attendi che lo stato Connesso venga visualizzato nella barra di stato. 3. Registra i dati sulle prestazioni: a. Nella barra degli strumenti, scegli Avvia registrazione. b. Esegui il test case e attendi il suo completamento. c. Scegli Stop Recording. 4. Salva e visualizza l'istantanea:	Sviluppatore di app

Operazione	Description	Competenze richieste
	a. Scegli Salva istantanea. b. Scegli la posizione in cui vuoi salvare l'istantanea, quindi scegli Salva.	

Operazione	Description	Competenze richieste
Analizza l' JProfiler istantanea per identificare i punti deboli in termini di prestazioni.	Segui questi passaggi per analizzare l'istantanea. JProfiler 1. Apri l'istantanea: a. Scegli File, Apri istantanea. b. Passa alla posizione dell'istantanea salvata. c. Scegliete Apri una nuova finestra per iniziare l'analisi. 2. Analizza le operazioni del database: a. Vai alla sezione JDBC/ Database e scegli la scheda JPA/Hibernate. b. Rivedi le query in base all'ora (che è l'opzione di ordinamento predefinita). c. Scegliete la query più in alto per l'ottimizzazione. 3. Verifica l'utilizzo della CPU: a. Nella sezione CPU, scegli la vista Hot Spots. b. Identifica il codice più in alto in termini di percentuale temporale per l'ottimizzazione. c. Annota eventuali schemi ricorrenti.	Sviluppatore di app

Operazione	Description	Competenze richieste
	4. Dopo aver identificato gli hotspot, stabilite una linea di base e quindi implement ate le strategie di ottimizza zione appropriate, come illustrato nelle seguenti epopee. Concentratevi su queste ottimizzazioni: • Per le query lente, usa SQLExecution Builder o l'ottimizzazione. DAOManager • Per un accesso frequente ai dati, implementa meccanismi di memorizza zione nella cache. • Per una logica aziendale complessa, semplifica la logica, evita di effettuar e troppe chiamate al database e determina se il codice può essere semplificato eseguendo una semplice query. Ad esempio, se il codice inserisce dati dalla Tabella A nella Tabella B, riscrivi questo codice per eseguire un comando SQL simile a: `INSERT INTO TABLE A SELECT * FROM TABLE B`	

Operazione	Description	Competenze richieste
	5. Continuate ad analizzare e ottimizzare il codice in ordine di impatto (dal più alto al più basso) finché l'applicazione non soddisfer à i requisiti prestazionali. Per ulteriori informazioni sull'utilizzo JProfiler, consulta la sezione Architettura e la JProfiler documentazione.	

Stabilisci una linea di base

Operazione	Description	Competenze richieste
Stabilisci una linea di base delle prestazioni prima di implementare le ottimizzazioni.	1. Come misurazione iniziale delle prestazioni, esegui il test case senza alcuna ottimizzazione. Crea un' JProfiler istantanea dello stato corrente e documenta le seguenti metriche: • Tempo totale di esecuzione • Numero di chiamate al database • Percentuale di utilizzo della CPU • Modello di utilizzo della memoria 2. Crea una linea di base del documento:	Sviluppatore di app

Operazione	Description	Competenze richieste
	a. Salva tutte le metriche con un timestamp e le condizioni di test. b. Prendi nota di eventuali fattori esterni che potrebbero influire sulle prestazioni. c. Archivia l' JProfiler istantanea per il confronto dopo l'ottimiz zazione. 3. Pianifica le modifiche di ottimizzazione descritte nella prossima epopea: a. Rivedi le metriche di base. b. Identifica l'operazione che desideri ottimizzare. c. Crea un backup del codice corrente. d. Documenta i miglioram enti previsti. 4. Misura ogni modifica prima dell'implementazione: a. Esegui un test case utilizzando le condizioni di base. b. Tempo di esecuzione record. c. Documenta il comportam ento attuale.	

Applica strategie di ottimizzazione

Operazione	Description	Competenze richieste
Ottimizza le chiamate di lettura.	Ottimizza il recupero dei dati utilizzando il DAOManager SetMaxResultsmetodo. Per ulteriori informazioni su questo approccio, consulta la sezione Architettura.	Sviluppatore di app, DAOManager
Rifattorizza la logica aziendale per evitare chiamate multiple al database.	Riduci le chiamate al database utilizzando una `JOIN` clausola SQL. Per dettagli ed esempi, consulta Ottimizzazione della logica aziendale nella sezione Architettura.	Sviluppatore di app, SQLExecution Builder
Rifattorizza il codice per utilizzare la memorizzazione nella cache per ridurre la latenza delle chiamate di lettura.	Per informazioni su questa tecnica, consulta la sezione Caching nella sezione Architettura.	Sviluppatore di app
Riscrivi codice inefficiente che utilizza più DAOManage r operazioni per semplici operazioni di aggiornamento.	Per ulteriori informazioni sull'aggiornamento dei dati direttamente nel database, consulta Ottimizzazione della logica aziendale nella sezione Architettura.	Sviluppatore di app

Verificate le strategie di ottimizzazione

Operazione	Description	Competenze richieste
Convalida ogni modifica di ottimizzazione in modo	1. Dopo aver implementato le ottimizzazioni:	Sviluppatore di app

Operazione	Description	Competenze richieste
iterativo mantenendo l'equival enza funzionale.	a. Esegui gli stessi casi di test partendo dalla linea di base che hai stabilito. b. Registra il nuovo orario di esecuzione. c. Verifica che i risultati corrispondano all'output di base. 2. se l'ottimizzazione porta a un miglioramento, documenta i guadagni effettivi, quindi procedi all'ottimizzazione successiv a. Se riscontri problemi, ripristina le modifiche, rivedi il tuo approccio e prova soluzioni alternative. 3. Tieni traccia dei progressi per ogni modifica: a. Confronta le nuove metriche con le metriche di base. b. Calcola la percentuale di miglioramento. c. Mantieni un totale costante di guadagni. **Note** L'utilizzo di metriche di base come riferimen	

Operazione	Description	Competenze richieste
	to garantisce una misurazione accurata dell'impatto di ogni ottimizzazione, mantenendo al contempo l'affidabilità del sistema.	

risoluzione dei problemi

Problema	Soluzione
Quando si esegue l'applicazione moderna, viene visualizzata un'eccezione con l'errore. `Query_ID not found`	Per risolvere il problema: 1. Verificate se `query.sql` esiste un file denominato nella `sql` cartella nella directory di lavoro dell'application server. 2. Verificate che il nome del `.sql` file corrispon da al nome del programma. Ad esempio, per un programma denominato`ABC`, il nome del file deve essere`ABC.sql`.
Hai aggiunto degli indici, ma non vedi alcun miglioramento delle prestazioni.	Segui questi passaggi per assicurarti che il motore di query utilizzi l'indice: 1. Se stai usando PostgreSQL, esegui il comando. `EXPLAIN <sql query>` 2. Controlla l'output del comando per o. `Seq Scan` `Index Scan` `Seq Scan`significa che il set di indici non è disponibile nelle colonne utilizzate dalla `where` clausola della query SQL.

Problema	Soluzione
	3. Crea un indice sul set di colonne corretto e prova a creare query che utilizzano l'indice creato.
Si verifica un' out-of-memoryeccezione.	Verificate che il codice rilasci la memoria contenuta nella struttura dei dati.
Le operazioni di scrittura in batch provocano la mancanza di record nella tabella	Esamina il codice per assicurarti che venga eseguita un'ulteriore operazione di scrittura quando il conteggio dei batch è diverso da zero.
La registrazione SQL non viene visualizzata nei registri delle applicazioni.	1. Verificate la posizione e la sintassi del file di configurazione della registrazione. 2. Assicuratevi che il livello di registro sia impostato su `DEBUG` per i pacchetti specifica ti. Esempio: `level.com.netfective.bluage .gapwalk.rt.blu4iv.dao: DEBUG level.org.springframework.jdbc .core: DEBUG` 3. Controllate le autorizzazioni dei file di registro del server delle applicazioni. 4. Riavviare il server delle applicazioni dopo le modifiche alla configurazione.

Risorse correlate

- Rifattorizzazione automatica delle applicazioni con AWS Blu Age (Guida per Modernizzazione del mainframe AWS l'utente)
- Documentazione pgAdmin
- JProfiler documentazione

Proteggi e semplifica l'accesso degli utenti in un database federativo Db2 su AWS utilizzando contesti affidabili

Sai Parthasaradhi, Amazon Web Services

Riepilogo

Molte aziende stanno migrando i propri carichi di lavoro mainframe legacy su Amazon Web Services (AWS). Questa migrazione include lo spostamento di IBM Db2 per z/OS database a Db2 per Linux, Unix e Windows (LUW) su Amazon Elastic Compute Cloud (Amazon). EC2 Durante una migrazione graduale da locale ad AWS, gli utenti potrebbero dover accedere ai dati in IBM Db2 z/OS e in Db2 LUW su Amazon EC2 fino alla completa migrazione di tutte le applicazioni e i database su Db2 LUW. In questi scenari di accesso remoto ai dati, l'autenticazione degli utenti può essere difficile perché piattaforme diverse utilizzano meccanismi di autenticazione diversi.

Questo modello illustra come configurare un server federativo su Db2 per LUW con Db2 come database remoto. z/OS Il modello utilizza un contesto affidabile per propagare l'identità di un utente da Db2 LUW a Db2 senza eseguire nuovamente l'autenticazione sul database z/OS remoto. Per ulteriori informazioni sui contesti affidabili, vedere la sezione Informazioni aggiuntive.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un'istanza Db2 in esecuzione su un'istanza Amazon EC2
- Un Db2 remoto per z/OS database in esecuzione in locale
- La rete locale connessa ad AWS tramite AWS Site-to-SiteVPN o AWS Direct Connect

Architecture

Architettura Target

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile nel cloud AWS. Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- AWS Site-to-Site VPN ti aiuta a trasferire il traffico tra le istanze che avvii su AWS e la tua rete remota.

Altri strumenti

- db2cli è il comando CLI (Interactive Command Line Interface) di Db2.

Epiche

Abilita la federazione sul database Db2 LUW in esecuzione su AWS

Operazione	Description	Competenze richieste
Abilita la federazione sul DB DB2 LUW.	Per abilitare la federazione su DB2 LUW, esegui il comando seguente. `update dbm cfg using federated YES`	DBA
Riavviare il database.	Per riavviare il database, esegui il comando seguente. `db2stop force; db2start;`	DBA

Catalogare il database remoto

Operazione	Description	Competenze richieste
Catalogare il z/OS sottosist ema Db2 remoto.	Per catalogare il z/OS database Db2 remoto su Db2 LUW in esecuzione su AWS,	DBA

Operazione	Description	Competenze richieste
	usa il seguente comando di esempio. `catalog TCPIP NODE tcpnode REMOTE mainframehost SERVER mainframeport`	
Catalogare il database remoto.	Per catalogare il database remoto, utilizzare il seguente comando di esempio. `catalog db dbnam1 as ndbnam1 at node tcpnode`	DBA

Crea la definizione del server remoto

Operazione	Description	Competenze richieste
Raccogli le credenziali utente per il database Db2 z/OS remoto.	Prima di procedere con i passaggi successivi, raccogli le seguenti informazioni: • Nome z/OS del sottosistema Db2: il nome Db2 z/OS catalogato su LUW nel passaggio precedente (ad esempio,) `ndbnam1` • Versione Db2: z/OS la versione del sottosistema Db2 z/OS (ad esempio,) `12` • ID z/OS utente Db2: l'utente con il privilegio BIND, necessario per creare solo	DBA

Operazione	Description	Competenze richieste
	la definizione del server (ad esempio,) `dbuser1` • z/OS Password Db2: la password per `dbuser1` (ad esempio,) `dbpasswd` • Utente z/OS proxy Db2: l'ID dell'utente proxy, che verrà utilizzato per stabilire una connessione affidabile (ad esempio,) `zproxy` • Password z/OS proxy Db2: la password per l'`zproxy`utente (ad esempio,) `zproxy`	
Crea il wrapper DRDA.	Per creare il wrapper DRDA, esegui il comando seguente. `CREATE WRAPPER DRDA;`	DBA

Operazione	Description	Competenze richieste
Crea la definizione del server.	Per creare la definizione del server, esegui il seguente comando di esempio. `CREATE SERVER ndbserver TYPE DB2/ZOS VERSION 12 WRAPPER DRDA AUTHORIZATION "dbuser1" PASSWORD "dbpasswd " OPTIONS ( DBNAME 'ndbnam1',FED_PROXY_USER 'ZPROXY' );` In questa definizione, `FED_PROXY_USER` specifica l'utente proxy che verrà utilizzato per stabilire connessioni affidabili al database Db2 z/OS . L'ID utente e la password di autorizzazione sono necessari solo per creare l'oggetto server remoto nel database Db2 LUW. Non verranno utilizzati in seguito durante il runtime.	DBA

Crea mappature degli utenti

Operazione	Description	Competenze richieste
Crea una mappatura utente per l'utente proxy.	Per creare una mappatura utente per l'utente proxy, esegui il comando seguente.	DBA

Operazione	Description	Competenze richieste
	`CREATE USER MAPPING FOR ZPROXY SERVER ndbserver OPTIONS (REMOTE_AUTHID 'ZPROXY', REMOTE_PASSWORD 'zproxy');`	
Crea mappature utente per ogni utente su Db2 LUW.	Crea mappature utente per tutti gli utenti del database Db2 LUW su AWS che devono accedere ai dati remoti tramite l'utente proxy. Per creare le mappature degli utenti, esegui il comando seguente. `CREATE USER MAPPING FOR PERSON1 SERVER ndbserver OPTIONS (REMOTE_AUTHID 'USERZID', USE_TRUSTED_CONTEXT 'Y');` L'istruzione specifica che un utente su Db2 LUW (`PERSON1`) può stabilire una connessione affidabile al database Db2 remoto (). z/OS `USE_TRUSTED_CONTEXT 'Y'` Dopo aver stabilito la connessione tramite l'utente proxy, l'utente può accedere ai dati utilizzando l'ID utente Db2 z/OS (). `REMOTE_AUTHID 'USERZID'`	DBA

Crea l'oggetto contestuale affidabile

Operazione	Description	Competenze richieste
Crea l'oggetto contestuale affidabile.	Per creare l'oggetto di contesto affidabile sul z/OS database Db2 remoto, utilizzate il seguente comando di esempio. `CREATE TRUSTED CONTEXT CTX_LUW_ZOS BASED UPON CONNECTION USING SYSTEM AUTHID ZPROXY ATTRIBUTES ( ADDRESS '10.10.10.10' ) NO DEFAULT ROLE ENABLE WITH USE FOR PUBLIC WITHOUT AUTHENTICATION;` In questa definizione, `CTX_LUW_ZOS` è un nome arbitrario per l'oggetto di contesto affidabile. L'oggetto contiene l'ID utente proxy e l'indirizzo IP del server da cui deve provenire la connessione affidabile. In questo esempio, il server è il database Db2 LUW su AWS. È possibile utilizzare il nome di dominio anziché l'indirizzo IP. La clausola `WITH USE FOR PUBLIC WITHOUT AUTHENTICATION` indica	DBA

Operazione	Description	Competenze richieste
	che la modifica dell'ID utente su una connessione affidabile è consentita per ogni ID utente. Non è necessario fornire una password.	

Risorse correlate

- Impianto di controllo degli accessi alle risorse IBM (RACF)
- Federazione IBM Db2 LUW
- Contesti affidabili

Informazioni aggiuntive

Contesti affidabili Db2

Un contesto affidabile è un oggetto di database Db2 che definisce una relazione di trust tra un server federato e un server di database remoto. Per definire una relazione di fiducia, il contesto affidabile specifica gli attributi di fiducia. Esistono tre tipi di attributi di fiducia:

- L'ID di autorizzazione del sistema che effettua la richiesta iniziale di connessione al database
- L'indirizzo IP o il nome di dominio da cui viene effettuata la connessione
- L'impostazione di crittografia per le comunicazioni di dati tra il server del database e il client del database

Una connessione affidabile viene stabilita quando tutti gli attributi di una richiesta di connessione corrispondono agli attributi specificati in qualsiasi oggetto di contesto affidabile definito sul server. Esistono due tipi di connessioni affidabili: implicite ed esplicite. Dopo aver stabilito una connessione implicita affidabile, un utente eredita un ruolo che non gli è disponibile al di fuori dell'ambito di tale definizione di connessione affidabile. Dopo aver stabilito una connessione affidabile esplicita, gli utenti possono attivare la stessa connessione fisica, con o senza autenticazione. Inoltre, agli utenti Db2 possono essere concessi ruoli che specificano privilegi utilizzabili solo all'interno della connessione affidabile. Questo modello utilizza una connessione esplicita e affidabile.

Contesto attendibile in questo modello

Una volta completato il pattern, PERSON1 su Db2 LUW accede ai dati remoti da Db2 utilizzando un contesto z/OS fidato federato. La connessione per PERSON1 viene stabilita tramite un utente proxy se la connessione proviene dall'indirizzo IP o dal nome di dominio specificato nella definizione del contesto affidabile. Dopo aver stabilito la connessione, PERSON1 l'ID z/OS utente Db2 corrispondente viene cambiato senza riautenticarsi e l'utente può accedere ai dati o agli oggetti in base ai privilegi Db2 impostati per quell'utente.

Vantaggi dei contesti fidati federati

- Questo approccio mantiene il principio del privilegio minimo eliminando l'uso di un ID utente o di un'applicazione comune che richiederebbe un superset di tutti i privilegi richiesti da tutti gli utenti.
- La vera identità dell'utente che esegue la transazione sia sul database federato che su quello remoto è sempre nota e può essere verificata.
- Le prestazioni migliorano perché la connessione fisica viene riutilizzata tra gli utenti senza che il server federato debba eseguire nuovamente l'autenticazione.

Trasferisci z/OS dati Db2 su larga scala su Amazon S3 in file CSV

Bruno Sahinoglu, Abhijit Kshirsagar e Ivan Schuster, Amazon Web Services

Riepilogo

Un mainframe è ancora un sistema di registrazione in molte aziende, che contiene un'enorme quantità di dati, tra cui entità di dati master con registrazioni delle transazioni commerciali correnti e storiche. Spesso è isolato e non è facilmente accessibile dai sistemi distribuiti all'interno della stessa azienda. Con l'avvento della tecnologia cloud e la democratizzazione dei big data, le aziende sono interessate a utilizzare le informazioni nascoste nei dati del mainframe per sviluppare nuove funzionalità aziendali.

Con questo obiettivo, le aziende stanno cercando di aprire i dati mainframe Db2 all'ambiente cloud Amazon Web Services (AWS). Le ragioni commerciali sono diverse e i metodi di trasferimento variano da caso a caso. Potresti preferire connettere l'applicazione direttamente al mainframe oppure replicare i dati quasi in tempo reale. Se il caso d'uso è quello di alimentare un data warehouse o un data lake, avere una up-to-date copia non è più un problema e la procedura descritta in questo schema potrebbe essere sufficiente, soprattutto se si desidera evitare i costi di licenza di prodotti di terze parti. Un altro caso d'uso potrebbe essere il trasferimento di dati su mainframe per un progetto di migrazione. In uno scenario di migrazione, i dati sono necessari per eseguire il test di equivalenza funzionale. L'approccio descritto in questo post è un modo conveniente per trasferire i dati Db2 all'ambiente cloud AWS.

Poiché Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è uno dei servizi AWS più integrati, puoi accedere ai dati da lì e raccogliere informazioni direttamente utilizzando altri servizi AWS come Amazon Athena, le funzioni AWS Lambda o Amazon. QuickSight Puoi anche caricare i dati su Amazon Aurora o Amazon DynamoDB utilizzando AWS Glue o AWS Database Migration Service (AWS DMS). Con questo obiettivo in mente, questo descrive come scaricare i dati Db2 in file CSV in formato ASCII sul mainframe e trasferire i file su Amazon S3.

A tal fine, sono stati sviluppati script mainframe per aiutare a generare linguaggi di controllo del lavoro (JCLs) per scaricare e trasferire tutte le tabelle Db2 necessarie.

Prerequisiti e limitazioni

Prerequisiti

- Un utente z/OS del sistema operativo IBM con autorizzazione a eseguire script Restructured Extended Executor (REXX) e JCL.

- Accesso a z/OS Unix System Services (USS) per generare chiavi private e pubbliche SSH (Secure Shell).
- Un bucket S3 scrivibile. Per ulteriori informazioni, consulta Crea il tuo primo bucket S3 nella documentazione di Amazon S3.
- Un server abilitato al protocollo SFTP (SSH File Transfer Protocol) di AWS Transfer Family che utilizza Service gestito come provider di identità e Amazon S3 come servizio di storage AWS. Per ulteriori informazioni, consulta Creare un server compatibile con SFTP nella documentazione di AWS Transfer Family.

Limitazioni

- Questo approccio non è adatto per la sincronizzazione dei dati quasi in tempo reale o in tempo reale.
- I dati possono essere spostati solo da Db2 z/OS ad Amazon S3, non viceversa.

Architecture

Stack tecnologico di origine

- Mainframe che esegue Db2 su z/OS

Stack tecnologico Target

- AWS Transfer Family
- Simple Storage Service (Amazon S3)
- Amazon Athena
- Amazon QuickSight
- AWS Glue
- Amazon Relational Database Service (Amazon RDS)
- Amazon Aurora
- Amazon Redshift

Architettura di origine e destinazione

Il diagramma seguente mostra il processo di generazione, estrazione e trasferimento di z/OS dati Db2 in formato ASCII CSV su un bucket S3.

1. Viene selezionato un elenco di tabelle per la migrazione dei dati dal catalogo Db2.
2. L'elenco viene utilizzato per guidare la generazione di lavori di scaricamento con le colonne numeriche e di dati in formato esterno.
3. I dati vengono quindi trasferiti su Amazon S3 utilizzando AWS Transfer Family.
4. Un job di estrazione, trasformazione e caricamento (ETL) di AWS Glue può trasformare i dati e caricarli in un bucket elaborato nel formato specificato, oppure AWS Glue può inserire i dati direttamente nel database.
5. Amazon Athena e Amazon QuickSight possono essere utilizzati per interrogare ed eseguire il rendering dei dati per favorire l'analisi.

Il diagramma seguente mostra un flusso logico dell'intero processo.

1. Il primo JCL, chiamato TABNAME, utilizzerà l'utilità Db2 DSNTIAUL per estrarre e generare l'elenco di tabelle che intendi scaricare da Db2. Per scegliere le tabelle, è necessario adattare manualmente l'input SQL per selezionare e aggiungere criteri di filtro per includere uno o più schemi Db2.
2. Il secondo JCL, chiamato REXXEXEC, utilizzerà uno scheletro JCL e il programma REXX forniti per elaborare l'elenco di tabelle creato da JCL TABNAME e generare un JCL per nome di tabella. Ogni JCL conterrà un passaggio per lo scaricamento della tabella e un altro passaggio per l'invio del file al bucket S3 utilizzando il protocollo SFTP.
3. L'ultimo passaggio consiste nell'eseguire JCL per scaricare la tabella e trasferire il file su AWS. L'intero processo può essere automatizzato utilizzando uno scheduler locale o su AWS.

Tools (Strumenti)

Servizi AWS

- Amazon Athena è un servizio di query interattivo che ti aiuta ad analizzare i dati direttamente in Amazon Simple Storage Service (Amazon S3) utilizzando SQL standard.
- Amazon Aurora è un motore di database relazionale completamente gestito creato per il cloud e compatibile con MySQL e PostgreSQL.

- AWS Glue è un servizio di estrazione, trasformazione e caricamento (ETL) completamente gestito. Ti aiuta a classificare, pulire, arricchire e spostare i dati in modo affidabile tra archivi di dati e flussi di dati.
- Amazon QuickSight è un servizio di business intelligence (BI) su scala cloud che ti aiuta a visualizzare, analizzare e riportare i tuoi dati in un'unica dashboard.
- Amazon Redshift è un servizio di data warehouse gestito su scala petabyte nel cloud AWS.
- Amazon Relational Database Service (Amazon RDS) ti aiuta a configurare, gestire e scalare un database relazionale nel cloud AWS.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Transfer Family è un servizio di trasferimento sicuro che consente di trasferire file da e verso i servizi di storage AWS.

Strumenti mainframe

- SSH File Transfer Protocol (SFTP) è un protocollo di trasferimento file sicuro che consente l'accesso remoto e il trasferimento di file tra server. SSH fornisce sicurezza crittografando tutto il traffico.
- DSNTIAUL è un programma di esempio fornito da IBM per lo scaricamento dei dati.
- DSNUTILB è un programma batch di utilità fornito da IBM per lo scarico di dati con diverse opzioni di DSNTIAUL.
- z/OS OpenSSH è una porta del software Open Source SSH in esecuzione su Unix System Service nella rete del sistema operativo IBM. z/OS. SSH is a secure, encrypted connection program between two computers running on a TCP/IP Fornisce diverse utilità, tra cui ssh-keygen.
- Lo script REXX (Restructured Extended Executor) viene utilizzato per automatizzare la generazione di JCL con i passaggi Db2 Unload e SFTP.

Codice

Il codice per questo pattern è disponibile nel repository unloaddb2. GitHub

Best practice

Per il primo scaricamento, i dati generati JCLs dovrebbero scaricare l'intera tabella.

Dopo il primo scaricamento completo, esegui scaricamenti incrementali per migliorare le prestazioni e risparmiare sui costi. Aggiorna la query SQL nel deck JCL del modello per adattare eventuali modifiche al processo di scaricamento.

È possibile convertire lo schema manualmente o utilizzando uno script su Lambda con Db2 SYSPUNCH come input. Per un processo industriale, AWS Schema Conversion Tool (SCT) è l'opzione preferita.

Infine, utilizza uno scheduler basato su mainframe o uno scheduler su AWS con un agente sul mainframe per gestire e automatizzare l'intero processo.

Epiche

Configura il bucket S3

Operazione	Description	Competenze richieste
Crea il bucket S3.	Per istruzioni, consulta Creare il tuo primo bucket S3.	Informazioni generali su AWS

Configurare il server Transfer Family

Operazione	Description	Competenze richieste
Crea un server compatibile con SFTP.	Per aprire e creare un server SFTP sulla console AWS Transfer Family, procedi come segue: 1. Nella pagina Scegli i protocolli, seleziona la casella di controllo SFTP (SSH File Transfer Protocol) — trasferimento di file tramite Secure Shell. 2. Per il provider di identità, scegli Servizio gestito.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	3. Per l'endpoint, scegli Accessibile pubblicamente. 4. Per il dominio, scegli Amazon S3. 5. Nella pagina Configura dettagli aggiuntivi, mantieni le impostazioni predefinite. 6. Crea il server.	
Crea un ruolo IAM per Transfer Family.	Per creare un ruolo AWS Identity and Access Management (IAM) per Transfer Family per accedere ad Amazon S3, segui le istruzioni in Creare un ruolo e una policy IAM.	Amministratore AWS
Aggiungi un utente gestito dal servizio Amazon S3.	Per aggiungere l'utente gestito dal servizio Amazon S3, segui le istruzioni nella documentazione AWS e usa il tuo ID utente mainframe.	Informazioni generali su AWS

Proteggi il protocollo di comunicazione

Operazione	Description	Competenze richieste
Crea la chiave SSH.	Nel tuo ambiente USS mainframe, esegui il seguente comando. `ssh-keygen -t rsa`	Sviluppatore di mainframe

Operazione	Description	Competenze richieste
	Note Quando viene richiesta una passphrase, lasciala vuota.	

Assegna i giusti livelli di autorizzazione alla cartella SSH e ai file chiave.	Per impostazione predefinita, le chiavi pubbliche e private verranno archiviate nella directory `/u/home/username/.ssh` dell'utente.
	È necessario fornire l'autorizzazione 644 ai file chiave e 700 alla cartella.
	`chmod 644 .ssh/id_rsa`
	`chmod 700 .ssh`
		Sviluppatore di mainframe

Operazione	Description	Competenze richieste
Copia il contenuto della chiave pubblica sul tuo utente gestito dal servizio Amazon S3.	Per copiare il contenuto della chiave pubblica generato da USS, apri la console AWS Transfer Family. 1. Nel riquadro di navigazione, selezionare Servers (Server). 2. Scegli l'identificatore nella colonna Server ID per visualizzare i dettagli del server 3. In Utenti, scegli un nome utente per visualizzare i dettagli dell'utente 4. In Chiavi pubbliche SSH, scegli Aggiungi chiave pubblica SSH per aggiungere la chiave pubblica a un utente. Per la chiave pubblica SSH, inserisci la tua chiave pubblica. La chiave viene convalidata dal servizio prima di poter aggiungere il nuovo utente. 5. Scegliere Add key (Aggiungi chiave).	Sviluppatore di mainframe

Genera il JCLs

Operazione	Description	Competenze richieste
Genera l'elenco di tabelle Db2 relativo all'ambito.	Fornisci un codice SQL di input per creare un elenco delle tabelle destinate alla migrazione dei dati. Questo passaggio richiede di specifica re i criteri di selezione per interrogare la tabella del catalogo Db2 SYSIBM.SY STABLES utilizzando una clausola SQL where. I filtri possono essere personalizzati per includere uno schema o nomi di tabelle specifici che iniziano con un prefisso particolare o basati su un timestamp per lo scaricame nto incrementale. L'output viene acquisito in un set di dati sequenziali fisici (PS) sul mainframe. Questo set di dati fungerà da input per la fase successiva della generazione di JCL. Prima di utilizzare JCL TABNAME (puoi rinominar lo se necessario), apporta le seguenti modifiche: 1. Sostituiscilo <Jobcard> con una classe di lavoro e un utente autorizzato a eseguire le utilità Db2.	Sviluppatore di mainframe

Operazione	Description	Competenze richieste

	2. Sostituisci < HLQ1 > o personalizza i nomi dei set di dati di output per soddisfare gli standard del tuo sito.
	3. Aggiorna lo stack STEPLIB di PDSEs (set di dati partizionato esteso) in base agli standard del tuo sito. L'esempio in questo modello utilizza i valori predefiniti IBM.
	4. Sostituisci il nome PLAN e LIB con i valori specifici dell'installazione.
	5. Sostituisci <Schema>e inserisci i <Prefix>tuoi criteri di selezione per il catalogo Db2.
	6. Salva il JCL risultante in una libreria PDS (partitio ned data set).
	7. Invia il JCL.
	
	Processo di estrazione dell'elenco delle tabelle Db2
	
	`<Jobcard>`
	`//*`
	`//* UNLOAD ALL THE TABLE`
	`NAMES FOR A PARTICULAR`
	`SCHEMA`
	`//*`
	`//STEP01 EXEC PGM=IEFBR`
	`14`	

Operazione	Description	Competenze richieste

	`//*`
	`//DD1 DD DISP=(MOD`
	`,DELETE,DELETE),`
	`// UNIT=SYSDA,`
	`// SPACE=(1000,`
	`(1,1)),`
	`// DSN=<HLQ1`
	`>.DSN81210.TABLIST`
	`//*`
	`//DD2 DD DISP=(MOD`
	`,DELETE,DELETE),`
	`// UNIT=SYSDA,`
	`// SPACE=(1000,`
	`(1,1)),`
	`// DSN=<HLQ1`
	`>.DSN81210.SYSPUNCH`
	`//*`
	`//UNLOAD EXEC PGM=IKJEF`
	`T01,DYNAMNBR=20`
	`//SYSTSPRT DD`
	` SYSOUT=*`
	`//STEPLIB DD`
	` DISP=SHR,DSN=DSNC1`
	`0.DBCG.SDSNEXIT`
	`// DD DISP=SHR,`
	`DSN=DSNC10.SDSNLOAD`
	`// DD DISP=SHR,`
	`DSN=CEE.SCEERUN`
	`// DD DISP=SHR,`
	`DSN=DSNC10.DBCG.RU`
	`NLIB.LOAD`
	`//SYSTSIN DD *`
	`  DSN SYSTEM(DBCG)`
	`  RUN PROGRAM(D`
	`SNTIAUL) PLAN(DSNT`
	`IB12) PARMS('SQL') -`
	`  LIB('DSNC`
	`10.DBCG.RUNLIB.LOAD')`
	`  END`
	`//SYSPRINT DD SYSOUT=*`
	`//*`
	`//SYSUDUMP DD SYSOUT=*`	

Operazione	Description	Competenze richieste

	//*
	//SYSREC00 DD DISP=(NEW
	,CATLG,DELETE),
	// UNIT=SYSD
	A,SPACE=(32760,(10
	00,500)),
	// DSN=<HLQ1
	>.DSN81210.TABLIST
	//*
	//SYSPUNCH DD DISP=(NEW
	,CATLG,DELETE),
	// UNIT=SYSD
	A,SPACE=(32760,(10
	00,500)),
	// VOL=SER=S
	CR03,RECFM=FB,LREC
	L=120,BLKSIZE=12
	// DSN=<HLQ1
	>.DSN81210.SYSPUNCH
	//*
	//SYSIN DD *
	  SELECT CHAR(CREA
	TOR), CHAR(NAME)
	  FROM SYSIBM.SY
	STABLES
	  WHERE OWNER =
	'<Schema>'
	  AND NAME LIKE
	'<Prefix>%'
	  AND TYPE = 'T';
	/*	

Operazione	Description	Competenze richieste
Modifica i modelli JCL.	I modelli JCL forniti con questo modello contengon o una scheda di lavoro generica e nomi di librerie. Tuttavia, la maggior parte dei siti mainframe avrà i propri standard di denominaz ione per i nomi dei set di dati, i nomi delle librerie e le job card. Ad esempio, potrebbe essere necessari a una classe di job specifica per eseguire i job Db2. Il Job Entry Subsystem implement a JES2 e JES3 può imporre ulteriori modifiche. Le librerie di caricamento standard potrebbero avere un primo qualificatore diverso SYS1 da, che è l'impostazione predefini ta di IBM. Pertanto, personali zza i modelli per tenere conto degli standard specifici del sito prima di eseguirli. Apportate le seguenti modifiche allo scheletro JCL UNLDSKEL: 1. Modifica la scheda lavoro con una classe di lavoro e un utente autorizzati a eseguire le utilità Db2. 2. Personalizza i nomi dei set di dati di output per	Sviluppatore di mainframe

Operazione	Description	Competenze richieste
	soddisfare gli standard del tuo sito. 3. Aggiorna lo stack STEPLIB in PDSEs base agli standard del tuo sito. L'esempio in questo modello utilizza i valori predefiniti IBM. 4. Sostituiscilo <DSN> con il nome del sottosistema Db2 e l'ID di correlazione. 5. Salvate il JCL risultante in una libreria PDS che fa parte dello stack ISPSLIB, che è la libreria di modelli scheletro standard per ISPF. Unload e SFTP (JCL skeleton) `//&USRPFX.U JOB (DB2UNLOAD),'JOB', CLASS=A,MSGCLASS=A, //  TIME=1440 ,NOTIFY=&USRPFX //* DELETE DATASETS //STEP01  EXEC PGM=IEFBR14 //DD01  DD DISP=(MOD ,DELETE,DELETE), //  UNIT=SYSD A, //  SPACE=(TR K,(1,1)), // DSN=&USRPFX..DB2.P UNCH.&JOBNAME`	

Operazione	Description	Competenze richieste
	`//DD02 DD DISP=(MOD ,DELETE,DELETE), // UNIT=SYSD A, // SPACE=(TR K,(1,1)), // DSN=&USRPFX..DB2.U NLOAD.&JOBNAME //* //* RUNNING DB2 EXTRACTION BATCH JOB FOR AWS DEMO //* //UNLD01 EXEC PGM=DSNUTILB,REGIO N=0M, // PARM='<DSN>,UNLOAD' //STEPLIB DD DISP=SHR,DSN=DSNC1 0.DBCG.SDSNEXIT // DD DISP=SHR, DSN=DSNC10.SDSNLOAD //SYSPRINT DD SYSOUT=* //UTPRINT DD SYSOUT=* //SYSOUT DD SYSOUT=* //SYSPUN01 DD DISP=(NEW,CATLG,DE LETE), // SPACE=(CY L,(1,1),RLSE), // DSN=&USRPFX..DB2.P UNCH.&JOBNAME //SYSREC01 DD DISP=(NEW,CATLG,DE LETE), // SPACE=(CY L,(10,50),RLSE), // DSN=&USRPFX..DB2.U NLOAD.&JOBNAME //SYSPRINT DD SYSOUT=* //SYSIN DD * UNLOAD`	

Operazione	Description	Competenze richieste

	<pre> DELIMITED COLDEL ','
	  FROM TABLE &TABNAME
	  UNLDDN SYSREC01
	  PUNCHDDN SYSPUN01
	  SHRLEVEL CHANGE
	 ISOLATION UR;
	 /*
	//*
	//* FTP TO AMAZON S3
	 BACKED FTP SERVER IF
	 UNLOAD WAS SUCCESSFUL
	//*
	//SFTP EXEC PGM=BPXBA
	TCH,COND=(4,LE),RE
	GION=0M
	//STDPARM DD *
	 SH cp "//'&USRP
	FX..DB2.UNLOAD.&JO
	BNAME'"
	   &TABNAME..csv;
	 echo "ascii
	 " >> uplcmd;
	 echo "PUT &TABNAME.
	.csv " >>>> uplcmd;
	 sftp -b uplcmd -i .ssh/
	id_rsa &FTPUSER.
	@&FTPSITE;
	 rm &TABNAME..csv;
	 //SYSPRINT DD SYSOUT=*
	 //STDOUT DD SYSOUT=*
	 //STDENV DD *
	 //STDERR DD SYSOUT=*</pre>	

Operazione	Description	Competenze richieste
Genera il Mass Unload JCL.	Questo passaggio prevede l'esecuzione di uno script REXX in un ambiente ISPF utilizzando JCL. Fornisci l'elenco delle tabelle relative all'ambito create nel primo passaggio come input per la generazione di massa di JCL in base al nome. `TABLIST DD` Il JCL genererà un nuovo JCL per nome di tabella in un set di dati partizionato specificato dall'utente specificato in base al nome. `ISPFILE DD` Alloca prima questa libreria. Ogni nuovo JCL avrà due passaggi: un passaggio per scaricare la tabella Db2 in un file e un passaggio per inviare il file al bucket S3. Apporta le seguenti modifiche in JCL REXXEXEC (puoi cambiare il nome): 1. Sostituitelo `Job card user ID` con un ID utente del mainframe che abbia l'autorità di scaricame nto sulle tabelle. Sostituis ciliSYSPROC,`ISPPLIB`, `ISPSLIBISPMLIB`, e `ISPTLIB` <HLQ1> valorizza li o personalizzali `DSN` per soddisfare gli standard del	Sviluppatore di mainframe

Operazione	Description	Competenze richieste
	tuo sito. Per scoprire i valori specifici dell'installazione, utilizzate il comando. `TSO ISRDDN` 2. Sostituiscilo `<MFUSER>` con un ID utente con privilegi di esecuzione del lavoro nell'installazione. 3. `<FTPUSER>` Sostituiscilo con un ID utente con privilegi USS e FTP nell'installazione. Si presume che questo ID utente e le relative chiavi di sicurezza SSH siano presenti nella directory Unix Systems Services appropriata sul mainframe. 4. Sostituiscilo `<AWS TransferFamily IP>` con l'indirizzo IP o il nome di dominio di AWS Transfer Family. Questo indirizzo verrà utilizzato per la fase SFTP. 5. Invia il JCL dopo aver richiesto la sistemazione standard del sito e aver aggiornato il programma REXX come descritto di seguito. Lavoro di generazione JCL di massa	

Operazione	Description	Competenze richieste

	`//RUNREXX JOB (CREATEJC`
	`L),'RUNS ISPF TABLIST',`
	`CLASS=A,MSGCLASS=A,`
	
	`//         TIME=1440`
	`,NOTIFY=&SYSUID`
	`//* Most of the values`
	` required can be updated`
	` to your site specific`
	`//* values using the`
	` command 'TSO ISRDDN' in`
	` your ISPF session.`
	`//* Update all the lines`
	` tagged with //update`
	` marker to desired`
	`//* site specific`
	` values.`
	`//ISPF EXEC PGM=IKJEF`
	`T01,REGION=2048K,D`
	`YNAMNBR=25`
	`//SYSPROC   DD`
	` DISP=SHR,DSN=USER.`
	`Z23D.CLIST`
	`//SYSEXEC   DD`
	` DISP=SHR,DSN=<HLQ1`
	`>.TEST.REXXLIB`
	`//ISPPLIB   DD`
	` DISP=SHR,DSN=ISP.S`
	`ISPPENU`
	`//ISPSLIB   DD`
	` DISP=SHR,DSN=ISP.S`
	`ISPSENU`
	`//          DD`
	` DISP=SHR,DSN=<HLQ1`
	`>.TEST.ISPSLIB`
	`//ISPMLIB   DD`
	` DSN=ISP.SISPMENU,D`
	`ISP=SHR`
	`//ISPTLIB   DD`
	` DDNAME=ISPTABL`	

Operazione	Description	Competenze richieste
	`// DD DSN=ISP.S ISPTENU,DISP=SHR //ISPTABL DD LIKE=ISP.SISPTENU, UNIT=VIO //ISPPROF DD LIKE=ISP.SISPTENU, UNIT=VIO //ISPLOG DD SYSOUT=*,RECFM=VA, LRECL=125 //SYSPRINT DD SYSOUT=* //SYSTSPRT DD SYSOUT=* //SYSUDUMP DD SYSOUT=* //SYSDBOUT DD SYSOUT=* //SYSTSPRT DD SYSOUT=* //SYSUDUMP DD SYSOUT=* //SYSDBOUT DD SYSOUT=* //SYSHELP DD DSN=SYS1.HELP,DISP =SHR //SYSOUT DD SYSOUT=* //* Input list of tablenames //TABLIST DD DISP=SHR,DSN=<HLQ1 >.DSN81210.TABLIST //* Output pds //ISPFILE DD DISP=SHR,DSN=<HLQ1 >.TEST.JOBGEN //SYSTSIN DD * ISPSTART CMD(ZSTEPS <MFUSER> <FTPUSER> <AWS TransferFamily IP>)`	

Operazione	Description	Competenze richieste
	`/*` Prima di utilizzare lo script REXX, apportate le seguenti modifiche: 1. Salvate lo script REXX in una libreria PDS definita sotto lo `SYSEXEC` stack nel JCL REXXEXEC modificato nel passaggio precedente e con ZSTEPS come nome del membro. Se vuoi rinominarlo, devi aggiornare il JCL per adattarlo alle tue esigenze. 2. Questo script utilizza l'opzione trace per stampare informazioni aggiuntive in caso di errori. È invece possibile aggiungere il `EXECIO` codice di gestione degli errori dopo `TSO` le istruzioni, e e rimuovere la riga di traccia. `ISPEXEC` 3. Questo script genera i nomi dei membri utilizzando la convenzione `L0Dnnnnn` di denominazione, che può supportare fino a 100.000 membri. Se hai più di 100.000 tabelle, usa un prefisso più breve e	

Operazione	Description	Competenze richieste
	modifica i numeri nell'istruzione. `tempjob` Script STEPS REX `/*REXX - - -

Operazione	Description	Competenze richieste
	/* member will be saved in ISPDSN library allocated in JCL */ ADDRESS ISPEXEC "FTCLOSE NAME("tempjob")" END ADDRESS TSO "FREE F(TABLIST) " ADDRESS TSO "FREE F(ISPFILE) " exit 0	

Esegui il JCLs

Operazione	Description	Competenze richieste
Eseguire la fase Db2 Unload.	Dopo la generazione di JCL, ne avrai tante JCLs quante sono le tabelle da scaricare. Questa storia utilizza un esempio generato da JCL per spiegare la struttura e i passaggi più importanti. Non è richiesta nessuna azione da parte tua. Le seguenti informazioni sono solo di riferimento. Se intendi inviare JCLs ciò che hai generato nel passaggio precedente, vai alla sezione Invia l' LODnnnnn JCLsattività.	Sviluppatore di mainframe, ingegnere di sistema

Operazione	Description	Competenze richieste
	Quando si scaricano dati Db2 utilizzando un JCL con l'utilità DSNUTILB Db2 fornita da IBM, è necessari o assicurarsi che i dati scaricati non contengano dati numerici compressi. A tale scopo`DELIMITED` , utilizzare il parametro DSNUTILB. Il `DELIMITED` parametro supporta lo scaricamento dei dati in formato CSV aggiungendo un carattere come delimitatore e virgolett e doppie per il campo di testo, rimuovendo la spaziatur a interna nella colonna VARCHAR e convertendo tutti i campi numerici in FORMATO ESTERNO, inclusi i campi DATE. L'esempio seguente mostra l'aspetto della fase di scaricamento nel JCL generato, utilizzando il carattere virgola come delimitatore. `UNLOAD DELIMITED COLDEL ',' FROM TABLE SCHEMA_NAME.TBNAME UNLDDN SYSREC01`	

Operazione	Description	Competenze richieste
	PUNCHDDN SYSPUN01 SHRLEVEL CHANGE ISOLATION UR;	

Operazione	Description	Competenze richieste
Esegui la fase SFTP.	Per utilizzare il protocollo SFTP di un JCL, utilizzate l'utilità BPXBATCH. L'utilità SFTP non può accedere direttamente ai set di dati MVS. È possibile utilizzare il comando copy (`cp`) per copiare il file `&USRPFX..DB2.UNLOAD.&JOBNAME` sequenzia le nella directory USS, dove diventa. `&TABNAME..csv` Esegui il `sftp` comando utilizzando la chiave privata (`id_rsa`) e utilizzando l'ID utente RACF come nome utente per connetterti all'indirizzo IP di AWS Transfer Family. `SH cp "//'&USRPFX..DB2.UNLOAD.&JOBNAME'" &TABNAME..csv; echo "ascii " >> uplcmd; echo "PUT &TABNAME..csv " >>>> uplcmd; sftp -b uplcmd -i .ssh/id_rsa &FTPUSER.@&FTP_TF_SITE; rm &TABNAME..csv;`	Sviluppatore di mainframe, ingegnere di sistema

Operazione	Description	Competenze richieste
Invia il. LODnnnnn JCLs	Il JCL precedente ha generato tutte le tabelle LODnnnnn JCL che devono essere scaricate, trasformate in CSV e trasferite nel bucket S3. Esegui il `submit` comando su tutto ciò che è stato generato. JCLs	Sviluppatore di mainframe, ingegnere di sistema

Risorse correlate

Per ulteriori informazioni sui diversi strumenti e soluzioni utilizzati in questo documento, consultate quanto segue:

- Guida per l'utente di z/OS OpenSSH
- Db2 z/OS — Esempi di istruzioni di controllo UNLOAD
- Db2 z/OS — Scaricamento di file delimitati
- Transfer Family: creazione di un server compatibile con SFTP
- Transfer Family: collaborazione con utenti gestiti dal servizio

Informazioni aggiuntive

Dopo aver archiviato i dati Db2 su Amazon S3, hai a disposizione molti modi per sviluppare nuove informazioni. Poiché Amazon S3 si integra con i servizi di analisi dei dati AWS, puoi utilizzare o esporre liberamente questi dati sul lato distribuito. Ad esempio, puoi eseguire le operazioni seguenti:

- Crea un data lake su Amazon S3 ed estrai informazioni preziose utilizzando query-in-place strumenti di analisi e apprendimento automatico senza spostare i dati.
- Avvia una funzione Lambda configurando un flusso di lavoro di elaborazione post-caricamento integrato con AWS Transfer Family.
- Sviluppa nuovi microservizi per accedere ai dati in Amazon S3 o in database completamente gestiti utilizzando AWS Glue, un servizio di integrazione dei dati senza server che semplifica la scoperta,

la preparazione e la combinazione di dati per l'analisi, l'apprendimento automatico e lo sviluppo di applicazioni.

In un caso di migrazione, poiché puoi trasferire qualsiasi dato dal mainframe a S3, puoi fare quanto segue:

- Elimina l'infrastruttura fisica e crea una strategia di archiviazione dei dati conveniente con Amazon S3 Glacier e S3 Glacier Deep Archive.
- Crea soluzioni di backup e ripristino scalabili, durevoli e sicure con Amazon S3 e altri servizi AWS, come S3 Glacier e Amazon Elastic File System (Amazon EFS), per aumentare o sostituire le funzionalità locali esistenti.

Altri modelli

- Implementa le automazioni di sicurezza per la AWS WAF soluzione utilizzando Terraform
- Replica i database mainframe su AWS utilizzando Precisly Connect

Gestione

Argomenti

- Gestione dei costi
- Informatica per l'utente finale
- High Performance Computing
- Cloud ibrido
- Gestione e governance
- Messaggi e comunicazioni
- Strategia multi-account

Gestione dei costi

Argomenti

- Crea report dettagliati su costi e utilizzo per i lavori AWS Glue utilizzando AWS Cost Explorer
- Crea report dettagliati su costi e utilizzo per i cluster Amazon EMR utilizzando AWS Cost Explorer
- Altri modelli

Crea report dettagliati su costi e utilizzo per i lavori AWS Glue utilizzando AWS Cost Explorer

Parijat Bhide e Aromal Raj Jayarajan, Amazon Web Services

Riepilogo

Questo modello mostra come tenere traccia dei costi di utilizzo dei processi di integrazione dei dati di AWS Glue configurando tag di allocazione dei costi definiti dall'utente. Puoi utilizzare questi tag per creare report dettagliati su costi e utilizzo in AWS Cost Explorer per lavori su più dimensioni. Ad esempio, puoi tenere traccia dei costi di utilizzo a livello di team, progetto o centro di costo.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Uno o più job AWS Glue con tag definiti dall'utente attivati

Architecture

Stack tecnologico Target

- AWS Glue
- AWS Cost Explorer

Il diagramma seguente mostra come applicare tag per tenere traccia dei costi di utilizzo per i lavori AWS Glue.

Il diagramma mostra il flusso di lavoro seguente:

1. Un ingegnere dei dati o un amministratore AWS crea tag di allocazione dei costi definiti dall'utente per i job AWS Glue.
2. Un amministratore AWS attiva i tag.
3. I tag inviano i metadati ad AWS Cost Explorer.

Tools (Strumenti)

- AWS Glue è un servizio di estrazione, trasformazione e caricamento (ETL) completamente gestito. Ti aiuta a classificare, pulire, arricchire e spostare i dati in modo affidabile tra archivi di dati e flussi di dati.
- AWS Cost Explorer ti aiuta a visualizzare e analizzare i costi e l'utilizzo di AWS.

Epiche

Crea e attiva tag per i tuoi lavori AWS Glue

Operazione	Description	Competenze richieste
Crea tag di allocazione dei costi definiti dall'utente per i tuoi job AWS Glue.	Per aggiungere tag a un job AWS Glue esistente 1. Accedi alla console di gestione AWS, quindi apri la console AWS Glue. 2. Nel riquadro di navigazione a sinistra, sotto ETL, scegli Jobs. 3. Nella sezione I tuoi lavori, scegli il nome del lavoro che stai taggando. 4. Seleziona la scheda Job details (Dettagli del processo). Quindi, espandi la sezione Proprietà avanzate. 5. Per Tag, scegli Aggiungi nuovo tag. 6. Per Chiave, inserisci un nome per il tag.	Ingegnere dei dati

Operazione	Description	Competenze richieste
	7. (Facoltativo) In Valore, inserite un valore da associare alla chiave. 8. (Facoltativo) Ripetete i passaggi 5-7 per ogni tag che desiderate creare per il lavoro. 9. Scegli Save (Salva). Per aggiungere tag a un nuovo lavoro AWS Glue 1. Crea un nuovo lavoro AWS Glue in base ai requisiti del tuo caso d'uso. Per istruzioni, consulta Working with jobs on the AWS Glue Console nella AWS Glue Developer Guide. 2. Quando configuri le impostazioni dei dettagli del lavoro, segui i passaggi 4-9 della sezione Aggiungere tag a un lavoro AWS Glue esistente di questa attività. **Note** Per ulteriori informazioni, consulta i tag AWS in AWS Glue nella AWS Glue Developer Guide.	

Operazione	Description	Competenze richieste
Attiva i tag di allocazione dei costi definiti dall'utente.	Segui le istruzioni in Attivazione dei tag di allocazione dei costi definiti dall'utente nella AWS Billing User Guide.	Amministratore AWS

Crea report su costi e utilizzo per i tuoi lavori in AWS Glue

Operazione	Description	Competenze richieste
Crea report sui costi e sull'utilizzo per i tuoi lavori AWS Glue utilizzando i filtri di tag in AWS Cost Explorer.	1. Accedi alla Console di gestione AWS e apri la console AWS Cost Management. 2. Nel riquadro di navigazione a sinistra, scegli Report. 3. Scegli Crea nuovo report. 4. Per Seleziona un tipo di rapporto, scegli Costo e utilizzo (consigliato). Quindi, scegli Crea rapporto. 5. Per Filtri, scegli Servizio. Viene visualizzato il menu a discesa Servizio. 6. Seleziona le caselle di controllo accanto a Glue. Quindi, scegli Applica filtri. 7. Per Filtri, scegli Tag. Viene visualizzato il menu a discesa Tag. 8. Scegli Team. Quindi, seleziona le caselle di controllo accanto ai team a cui hai assegnato i tag.	AWS generale, amministratore AWS

Operazione	Description	Competenze richieste
	Escludi tutti i team a cui non hai assegnato tag. Quindi, scegli Applica filtri. 9. Nella parte superiore del grafico, scegli Tag. Quindi, scegli i tag per i job AWS Glue per i quali desideri creare un report. 10. Nella parte superiore del grafico, scegli il menu a discesa Ultimi 3 mesi e scegli l'intervallo di tempo che desideri coprire nel rapporto. Quindi, scegli il menu a discesa Mensile e scegli come desideri che le voci del rapporto vengano aggregate in base al periodo di tempo. 11. Selezionare Save as (Salva con nome). Quindi, inserisci un titolo per il rapporto. 12. Scegli Salva rapporto. Per ulteriori informazioni, consulta Exploring your data using Cost Explorer nella AWS Cost Management User Guide.	

Crea report dettagliati su costi e utilizzo per i cluster Amazon EMR utilizzando AWS Cost Explorer

Parijat Bhide e Aromal Raj Jayarajan, Amazon Web Services

Riepilogo

Questo modello mostra come tenere traccia dei costi di utilizzo dei cluster Amazon EMR configurando tag di allocazione dei costi definiti dall'utente. Puoi utilizzare questi tag per creare report dettagliati su costi e utilizzo in AWS Cost Explorer per cluster su più dimensioni. Ad esempio, puoi tenere traccia dei costi di utilizzo a livello di team, progetto o centro di costo.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Uno o più cluster EMR con tag definiti dall'utente attivati

Architecture

Stack tecnologico Target

- Amazon EMR
- AWS Cost Explorer

Architettura di destinazione

Il diagramma seguente mostra come applicare tag per tenere traccia dei costi di utilizzo per cluster Amazon EMR specifici.

Il diagramma mostra il flusso di lavoro seguente:

1. Un ingegnere dei dati o un amministratore AWS crea tag di allocazione dei costi definiti dall'utente per i cluster Amazon EMR.
2. Un amministratore AWS attiva i tag.
3. I tag inviano i metadati ad AWS Cost Explorer.

Tools (Strumenti)

Strumenti

- Amazon EMR è una piattaforma di cluster gestita che semplifica l'esecuzione di framework di big data su AWS per elaborare e analizzare grandi quantità di dati.
- AWS Cost Explorer ti aiuta a visualizzare e analizzare i costi e l'utilizzo di AWS.

Epiche

Crea e attiva tag per i tuoi cluster Amazon EMR

Operazione	Description	Competenze richieste
Crea tag di allocazione dei costi definiti dall'utente per i tuoi cluster Amazon EMR.	Per aggiungere tag a un cluster Amazon EMR esistente Segui le istruzioni in Aggiungere tag a un cluster esistente nella Amazon EMR Management Guide. Per aggiungere tag a un nuovo cluster Amazon EMR Segui le istruzioni in Aggiungere tag a un nuovo cluster nella Amazon EMR Management Guide. Per ulteriori informazioni su come configurare un cluster Amazon EMR, consulta Pianificare e configurare i cluster nella Amazon EMR Management Guide.	Ingegnere dei dati
Attiva i tag di allocazione dei costi definiti dall'utente.	Segui le istruzioni in Attivazione dei tag di allocazione dei	Amministratore AWS

Operazione	Description	Competenze richieste
	costi definiti dall'utente nella AWS Billing User Guide.	

Crea report su costi e utilizzo per i tuoi cluster Amazon EMR

Operazione	Description	Competenze richieste
Crea report su costi e utilizzo per i tuoi cluster Amazon EMR utilizzando i filtri di tag in AWS Cost Explorer.	1. Accedi alla Console di gestione AWS e apri la console AWS Cost Management. 2. Nel riquadro di navigazione a sinistra, scegli Report. 3. Scegli Crea nuovo report. 4. Per Seleziona un tipo di rapporto, scegli Costo e utilizzo (consigliato). Quindi, scegli Crea rapporto. 5. Per Filtri, scegli Servizio. Viene visualizzato il menu a discesa Servizio. 6. Seleziona le caselle di controllo accanto a EMR (Elastic MapReduce) e EC2-Instances (Elastic Compute Cloud — Compute). Quindi, scegli Applica filtri. 7. Per Filtri, scegli Tag. Viene visualizzato il menu a discesa Tag. 8. Scegli Squadra. Quindi, seleziona le caselle di	AWS generale, amministratore AWS

Operazione	Description	Competenze richieste
	controllo accanto ai team a cui hai assegnato i tag. Escludi tutti i team a cui non hai assegnato tag. Quindi, scegli Applica filtri. 9. Nella parte superiore del grafico, scegli Tag. Quindi, scegli i tag per i cluster Amazon EMR per i quali desideri creare un report. 10. Nella parte superiore del grafico, scegli il menu a discesa Ultimi 3 mesi e scegli l'intervallo di tempo che desideri coprire il rapporto. Quindi, scegli il menu a discesa Mensile e scegli come desideri che le voci del rapporto vengano aggregate in base al periodo di tempo. 11. Selezionare Save as (Salva con nome). Quindi, inserisci un titolo per il rapporto. 12. Scegli Salva rapporto. Per ulteriori informazioni, consulta Exploring your data using Cost Explorer nella AWS Cost Management User Guide.	

Altri modelli

- Automatizza le operazioni AWS dell'infrastruttura utilizzando Amazon Bedrock
- Inventaria automaticamente AWS le risorse su più account e regioni
- Automatizza la creazione di risorse Amazon WorkSpaces Applications utilizzando AWS CloudFormation
- Archivia automaticamente gli elementi su Amazon S3 utilizzando DynamoDB TTL
- Arresta e avvia automaticamente un'istanza database Amazon RDS utilizzando AWS Systems Manager Maintenance Windows
- Crea report dettagliati su costi e utilizzo per Amazon RDS e Amazon Aurora
- Stima dei costi di storage per una tabella Amazon DynamoDB
- Stima del costo di una tabella DynamoDB per la capacità su richiesta
- Configura la scalabilità automatica basata sugli eventi in Amazon EKS utilizzando Amazon EKS Pod Identity e KEDA
- Coordina la dipendenza dalle risorse e l'esecuzione delle attività utilizzando il costrutto AWS Fargate WaitCondition hook

Informatica per l'utente finale

Argomenti

- Implementa l'autenticazione SAML 2.0 per Amazon WorkSpaces utilizzando Auth0 e AWS Managed Microsoft AD
- Altri modelli

Implementa l'autenticazione SAML 2.0 per Amazon WorkSpaces utilizzando Auth0 e AWS Managed Microsoft AD

Siva Vinnakota e Shantanu Padhye, Amazon Web Services

Riepilogo

Questo modello esplora come integrare Auth0 AWS Directory Service for Microsoft Active Directory per creare una solida soluzione di autenticazione SAML 2.0 per il tuo ambiente Amazon. WorkSpaces Spiega come stabilire una federazione tra questi sistemi Servizi AWS per abilitare funzionalità avanzate come l'autenticazione a più fattori (MFA) e i flussi di accesso personalizzati, preservando al contempo l'accesso al desktop senza interruzioni. AWS Managed Microsoft AD Che tu gestisca solo una manciata di utenti o migliaia, questa integrazione aiuta a fornire flessibilità e sicurezza alla tua organizzazione. Questo modello fornisce i passaggi per il processo di configurazione in modo da poter implementare questa soluzione nel proprio ambiente.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- AWS Managed Microsoft AD
- Un desktop fornito in Amazon WorkSpaces Personal associato a AWS Managed Microsoft AD
- Un'istanza Amazon Elastic Compute Cloud (Amazon EC2)
- Un account Auth0

Limitazioni

Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Architecture

Il processo di autenticazione SAML 2.0 per un'applicazione WorkSpaces client consiste in cinque passaggi illustrati nel diagramma seguente. Questi passaggi rappresentano un tipico flusso di lavoro per l'accesso. È possibile utilizzare questo approccio distribuito all'autenticazione dopo aver seguito

le istruzioni riportate in questo schema, per fornire un metodo strutturato e sicuro per l'accesso degli utenti.

Flusso di lavoro:

1. Registrazione. L'utente avvia l'applicazione client WorkSpaces e inserisce il codice di WorkSpaces registrazione per la propria directory abilitata per SAML WorkSpaces . WorkSpaces restituisce l'URL del provider di identità (IdP) Auth0 all'applicazione client.
2. Accedi.Il WorkSpaces client reindirizza al browser Web dell'utente utilizzando l'URL Auth0. L'utente si autentica con nome utente e password. Auth0 restituisce un'asserzione SAML al browser del client. L'asserzione SAML è un token crittografato che afferma l'identità dell'utente.
3. Autenticazione. Il browser del client pubblica l'asserzione SAML sull' Accedi ad AWS endpoint per convalidarla. Accedi ad AWS consente al chiamante di assumere un AWS Identity and Access Management ruolo (IAM). Ciò restituisce un token che contiene credenziali temporanee per il ruolo IAM.
4. WorkSpaces accesso. Il WorkSpaces client presenta il token all'endpoint del WorkSpaces servizio. WorkSpaces scambia il token con un token di sessione e restituisce il token di sessione al WorkSpaces client con un URL di accesso. Quando il WorkSpaces client carica la pagina di accesso, il valore del nome utente viene compilato con il `NameId` valore passato nella risposta SAML.
5. Streaming. L'utente inserisce la propria password e si autentica nella WorkSpaces directory. Dopo l'autenticazione, WorkSpaces restituisce un token al client. Il client reindirizza nuovamente al WorkSpaces servizio e presenta il token. Questo negozia una sessione di streaming tra il WorkSpaces client e il. WorkSpace

> **Note**
>
> Per configurare un'esperienza Single Sign-On senza interruzioni che non richieda la richiesta di una password, consulta le sezioni Autenticazione basata su certificati e Personale nella documentazione. WorkSpaces WorkSpaces

Tools (Strumenti)

Servizi AWS

- Amazon WorkSpaces è un servizio di infrastruttura desktop virtuale (VDI) completamente gestito che fornisce agli utenti desktop basati sul cloud senza dover procurarsi e distribuire hardware o installare software complessi.
- AWS Directory Service for Microsoft Active Directoryconsente ai carichi di lavoro e alle risorse compatibili con le directory di AWS utilizzare Microsoft Active Directory in. Cloud AWS

Altri strumenti

- Auth0 è una piattaforma di autenticazione e autorizzazione che ti aiuta a gestire l'accesso alle tue applicazioni.

Epiche

Configura il connettore LDAP di Active Directory in Auth0

Operazione	Description	Competenze richieste
Installa il connettore LDAP di Active Directory in Auth0 con. AWS Managed Microsoft AD	1. Accedi alla dashboard Auth0 e scegli Authentic ation, Enterprise, Active Directory/LDAP. Scegli Crea una connessione. 2. Fornisci il nome per la tua connessione ad Active Directory, quindi scegli Crea. 3. Nella scheda Configura zione, scarica l'agente per il tuo sistema operativo. Una volta completata l'installazione, il browser predefinito visualizza una richiesta per l'URL del ticket.	Amministratore del cloud, architetto del cloud

Operazione	Description	Competenze richieste
	4. Inserisci l'URL del ticket di provisioning, che deve essere univoco per la stringa di connessione, quindi scegli Continua. 5. Nella finestra di dialogo di configurazione AD LDAP, in Nome utente e Password, inserisci le credenziali di amministratore, quindi scegli Salva. Una volta stabilita la connessione a Auth0, il registro di configurazione viene visualizzato con lo stato di segnalazione OK per tutti i controlli.	

Operazione	Description	Competenze richieste
Crea un'applicazione in Auth0 per generare il file manifest dei metadati SAML.	1. Accedi alla dashboard Auth0 e crea una nuova applicazione seguendo le istruzioni nella documentazione di Auth0. 2. Nella dashboard Auth0, scegli il nome dell'applicazione per accedere alle relative impostazioni di configurazione. Nella AddOnsscheda, scegli App SAML2 Web. 3. Nella scheda Impostazioni del componente aggiuntivo, per Application Callback URL, inserisci https://signin.aws.amazon.com/saml. Qui è dove il token SAML invierà la richiesta. POST 4. Nella scheda Impostazioni, nella casella Impostazioni, incolla il seguente codice di configurazione SAML in formato JSON: `{ "audience": "https://signin.aws.amazon.com/saml", "mappings": { "email": "http://schemas.xmlsoap.org/ws/2005/05/identi`	Amministratore del cloud, architetto del cloud

Operazione	Description	Competenze richieste
	ty/claims/sAMAccou ntName", "name": "http://s chemas.xmlsoap.org/ ws/2005/05/identity/ claims/name" }, "createUpnClaim": false, "passthroughClaims WithNoMapping": false, "mapUnknownClaimsA sIs": false, "mapIdentities": false, "nameIdentifierFor mat": "urn:oasi s:names:tc:SAML:2. 0:nameid-format:pe rsistent", "nameIdentifierPro bes": ["http://s chemas.auth0.com/s AMAccountName"] } 5. Salva, quindi scegli Abilita. 6. Scegli la scheda Utilizzo e scarica il file manifesto dei metadati per il provider di identità. Queste informazi oni sono necessarie per i passaggi successivi. 7. Chiudi la finestra dell'app SAML2 Web.	

Operazione	Description	Competenze richieste
	8. Nella schermata delle applicazioni, scegli Connessioni. In Enterprise, seleziona il Directory/LDAP connettore attivo corretto e abilitalo.	

Configura IdP, ruolo e policy per SAML 2.0 in IAM

Operazione	Description	Competenze richieste
Crea un IdP SAML 2.0 in IAM.	Per configurare SAML 2.0 come IdP, segui i passaggi descritti in Creare un provider di identità SAML in IAM nella documentazione IAM.	Amministratore cloud
Crea un ruolo e una policy IAM per la federazione SAML 2.0.	1. Crea un ruolo IAM per la federazione SAML 2.0. Per istruzioni, consulta il passaggio 2 delle istruzioni per la configurazione di SAML 2.0 for WorkSpaces Personal nella WorkSpaces documentazione. 2. Crea una policy IAM e associala al ruolo creato nel passaggio precedente. Per istruzioni, consulta il passaggio 3 nelle istruzioni per la configurazione di SAML 2.0 for WorkSpaces Personal nella WorkSpaces documentazione.	Amministratore cloud

Configura le asserzioni in Auth0

Operazione	Description	Competenze richieste
Configura le asserzioni Auth0 e SAML.	Puoi utilizzare le azioni Auth0 per configurare le asserzioni nelle risposte SAML 2.0. Un'asserzione SAML è un token crittografato che afferma l'identità dell'utente. 1. Accedi alla dashboard Auth0. Scegli Actions, Library, Create Action, Build da zero. 2. Fornisci i seguenti valori, quindi scegli Crea. Nome: specifica un nome per l'azione Trigger: scegli Login/Post Login Runtime: scegli il nodo 18 3. Nella schermata successiva, inserisci il seguente codice: `exports.onExecutePostLogin = async (event, api) => { if (event.client.name === "Workspace_Saml") { const awsRole = 'arn:aws:iam::030784294031:role/Workspace_Auth0,arn:aw`	Amministratore cloud

Operazione	Description	Competenze richieste

	s:iam::030784294031:saml-provider/Auth0';
	const awsRoleSession = event.user.sAMAccountName;
	const email = event.user.emails[0];
	api.samlResponse.setDestination('https://signin.aws.amazon.com/saml');
	api.samlResponse.setAttribute('https://aws.amazon.com/SAML/Attributes/Role', awsRole)
	api.samlResponse.setAttribute('https://aws.amazon.com/SAML/Attributes/RoleSessionName', awsRoleSession)
	api.samlResponse.setAttribute('https://aws.amazon.com/SAML/Attributes/PrincipalTag:Email', email)
	} return;
	};
	4. Seleziona Implementa.
	
	Questo completa la configurazione dell'autenticazione SAML 2.0 per i desktop	

Operazione	Description	Competenze richieste
	WorkSpaces personali. La sezione Architettura illustra il processo di autenticazione dopo la configurazione.	

risoluzione dei problemi

Problema	Soluzione
Problemi di autenticazione SAML 2.0 in WorkSpaces	Se riscontri problemi durante l'implementazione dell'autenticazione SAML 2.0 for WorkSpace s Personal, segui i passaggi e i link descritti nell'articolo di AWS re:POST sulla risoluzione dei problemi di autenticazione SAML 2.0. Per ulteriori informazioni sull'analisi degli errori SAML 2.0 durante l'accesso, consulta: WorkSpaces • Come posso creare un file HAR e i registri della console dal mio browser per un caso? Supporto AWS (AWS re:POST) • Visualizza una risposta SAML nel tuo browser (documentazione IAM)

Risorse correlate

- Configura SAML 2.0 for WorkSpaces Personal (documentazione) WorkSpaces
- Documentazione Auth0

Altri modelli

- Automatizza la creazione di risorse Amazon WorkSpaces Applications utilizzando AWS CloudFormation
- Migliora la qualità delle chiamate sulle postazioni di lavoro degli agenti nei contact center Amazon Connect
- Esegui le attività di AWS Systems Manager Automation in modo sincrono da AWS Step Functions

High Performance Computing

Argomenti

- Implementa un file system Lustre per l'elaborazione dei dati ad alte prestazioni utilizzando Terraform e DRA
- Configura una dashboard di monitoraggio Grafana per AWS ParallelCluster
- Altri modelli

Implementa un file system Lustre per l'elaborazione dei dati ad alte prestazioni utilizzando Terraform e DRA

Arun Bagal e Ishwar Chauthaiwale, Amazon Web Services

Riepilogo

Questo modello implementa automaticamente un file system Lustre AWS e lo integra con Amazon Elastic Compute Cloud (Amazon) EC2 e Amazon Simple Storage Service (Amazon S3).

Questa soluzione consente di configurare rapidamente un ambiente di elaborazione ad alte prestazioni (HPC) con storage integrato, risorse di elaborazione e accesso ai dati Amazon S3. Combina le funzionalità di storage di Lustre con le opzioni di elaborazione flessibili fornite da Amazon EC2 e lo storage di oggetti scalabile di Amazon S3, in modo da poter affrontare carichi di lavoro ad alta intensità di dati in machine learning, HPC e analisi dei big data.

Il modello utilizza un modulo HashiCorp Terraform e Amazon FSx for Lustre per semplificare il seguente processo:

- Fornitura di un file system Lustre
- Creazione di un'associazione di repository di dati (DRA) tra FSx for Lustre e un bucket S3 per collegare il file system Lustre agli oggetti Amazon S3
- Creazione di un'istanza EC2
- Montaggio del file system Lustre con il DRA collegato ad Amazon S3 sull'istanza EC2

I vantaggi di questa soluzione includono:

- Design modulare. È possibile gestire e aggiornare facilmente i singoli componenti di questa soluzione.
- Scalabilità. È possibile implementare rapidamente ambienti coerenti tra Account AWS le nostre regioni.
- Flessibilità. È possibile personalizzare l'implementazione in base alle proprie esigenze specifiche.
- Le migliori pratiche. Questo modello utilizza moduli preconfigurati che seguono le AWS migliori pratiche.

Per ulteriori informazioni sui file system Lustre, consultate il sito Web Lustre.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Una politica di privilegi minimi AWS Identity and Access Management (IAM) (vedi istruzioni)

Limitazioni

FSx for Lustre limita il file system Lustre a una singola zona di disponibilità, il che potrebbe essere un problema se si hanno requisiti di elevata disponibilità. Se la zona di disponibilità che contiene il file system si guasta, l'accesso al file system viene perso fino al ripristino. Per ottenere un'elevata disponibilità, puoi utilizzare DRA per collegare il file system Lustre ad Amazon S3 e trasferire dati tra zone di disponibilità.

Versioni del prodotto

- Terraform versione 1.9.3 o successiva
- HashiCorp AWS Provider versione 4.0.0 o successiva

Architecture

Il diagramma seguente mostra l'architettura di FSx for Lustre e quella complementare in. Servizi AWS Cloud AWS

L'architettura include quanto segue:

- Un bucket S3 viene utilizzato come posizione di archiviazione durevole, scalabile ed economica per i dati. L'integrazione tra FSx for Lustre e Amazon S3 fornisce un file system ad alte prestazioni perfettamente collegato ad Amazon S3.
- FSx for Lustre esegue e gestisce il file system Lustre.
- Amazon CloudWatch Logs raccoglie e monitora i dati di log dal file system. Questi log forniscono informazioni dettagliate sulle prestazioni, sullo stato e sull'attività del file system Lustre.
- Amazon EC2 viene utilizzato per accedere ai file system Lustre utilizzando il client Lustre open source. EC2 le istanze possono accedere ai file system da altre zone di disponibilità all'interno dello stesso cloud privato virtuale (VPC). La configurazione di rete consente l'accesso attraverso

sottoreti all'interno del VPC. Dopo aver montato il file system Lustre sull'istanza, puoi lavorare con i relativi file e directory proprio come faresti con un file system locale.

- AWS Key Management Service (AWS KMS) migliora la sicurezza del file system fornendo la crittografia dei dati inattivi.

Automazione e scalabilità

Terraform semplifica l'implementazione, la gestione e la scalabilità dei file system Lustre in più ambienti. In FSx Lustre, un singolo file system ha limiti di dimensione, quindi potrebbe essere necessario ridimensionarlo orizzontalmente creando più file system. Puoi utilizzare Terraform per fornire più file system Lustre in base alle tue esigenze di carico di lavoro.

Tools (Strumenti)

Servizi AWS

- Amazon CloudWatch Logs ti aiuta a centralizzare i log di tutti i tuoi sistemi e applicazioni, Servizi AWS così puoi monitorarli e archiviarli in modo sicuro.
- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon FSx for Lustre semplifica ed economica l'avvio, l'esecuzione e la scalabilità di un file system Lustre ad alte prestazioni.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Archivio di codice

Il codice per questo modello è disponibile nel file system GitHub Provision FSx for Lustre utilizzando il repository Terraform.

Best practice

- Le seguenti variabili definiscono il file system Lustre. Assicurati di configurarle correttamente in base al tuo ambiente, come indicato nella sezione Epics.

- `storage_capacity`— La capacità di archiviazione del file system Lustre, in. GiBs L'impostazione minima e predefinita è 1200 GiB.
- `deployment_type`— Il tipo di distribuzione per il file system Lustre. Per una spiegazione delle due opzioni `PERSISTENT_1` e `PERSISTENT_2` (impostazione predefinita), consultate la documentazione FSx di Lustre.
- `per_unit_storage_throughput`— La velocità di lettura e scrittura, in secondi MBs per TiB.
- `subnet_id`— L'ID della sottorete privata in cui si desidera implementare FSx Lustre.
- `vpc_id`— L'ID del tuo cloud privato virtuale su AWS cui desideri implementare FSx Lustre.
- `data_repository_path`— Il percorso del bucket S3 che sarà collegato al file system Lustre.
- `iam_instance_profile`— Il profilo dell'istanza IAM da utilizzare per avviare l'istanza. EC2
- `kms_key_id`— L'Amazon Resource Name (ARN) della AWS KMS chiave che verrà utilizzata per la crittografia dei dati.

- Garantisci l'accesso e il posizionamento corretti alla rete all'interno del VPC utilizzando le variabili `security_group` and`vpc_id`.
- Esegui il `terraform plan` comando come descritto nella sezione Epics per visualizzare in anteprima e verificare le modifiche prima di applicarle. Questo aiuta a catturare potenziali problemi e assicura che tu sia a conoscenza di ciò che verrà distribuito.
- Usa il `terraform validate` comando come descritto nella sezione Epics per verificare la presenza di errori di sintassi e confermare che la configurazione sia corretta.

Epiche

Configurare l'ambiente

Operazione	Description	Competenze richieste
Installa Terraform.	Per installare Terraform sul tuo computer locale, segui le istruzioni nella documenta zione di Terraform.	AWS DevOps, DevOps ingegnere
Configura AWS le credenziali.	Per configurare il profilo AWS Command Line Interface (AWS CLI) per l'account,	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	segui le istruzioni nella AWS documentazione.	
Clona il GitHub repository.	Per clonare il GitHub repository, esegui il comando: `git clone https://github.com/aws-samples/provision-fsx-lustre-with-terraform.git`	AWS DevOps, DevOps ingegnere

Configura e distribuisci FSx per Lustre

Operazione	Description	Competenze richieste
Aggiorna la configurazione di distribuzione.	1. Nel repository clonato sul computer locale, accedi alla directory: `fsx_deployment` `cd fsx_deployment` 2. Apri il `terraform.tfvars` file e aggiorna i valori delle seguenti variabili: • `vpc_id` • `subnet_id` • `data_repository_path` • `iam_instance_profile` • `kms_key_id`	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
	Per le descrizioni di queste variabili, consulta la sezione Best practice. 3. Nella stessa directory, apri il `locals.tf` file e aggiorna gli intervall i CIDR per `fsx_inres s` le variabili del gruppo `fsx_egress` di sicurezza. 4. Se necessario, apri il `variables.tf` file e aggiorna i valori predefiniti di queste variabili: • `storage_capacity` • `deployment_type` • `per_unit_storage_t hroughput` Per le descrizioni di queste variabili, consulta la sezione Best practice.	
Inizializza l'ambiente Terraform.	Per inizializzare il tuo ambiente per eseguire il modulo Terraform`fsx_deplo yment` , esegui: `terraform init`	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste
Convalida la sintassi Terraform.	Per verificare la presenza di errori di sintassi e confermar e che la configurazione sia corretta, esegui: `terraform validate`	AWS DevOps, DevOps ingegnere
Convalida la configurazione Terraform.	Per creare un piano di esecuzione Terraform e visualizzare in anteprima la distribuzione, esegui: `terraform plan -var-file terraform.tfvars`	AWS DevOps, DevOps ingegnere
Implementa il modulo Terraform.	Per distribuire le risorse FSx for Lustre, esegui: `terraform apply -var-file terraform.tfvars`	AWS DevOps, DevOps ingegnere

Pulisci AWS le risorse

Operazione	Description	Competenze richieste
Rimuovi AWS risorse.	Dopo aver finito di utilizzar e l'ambiente FSx for Lustre, puoi rimuovere le AWS risorse distribuite da Terraform per evitare di incorrere in costi inutili. Il modulo Terraform fornito nel repository del codice automatizza questa pulizia.	AWS DevOps, DevOps ingegnere

Operazione	Description	Competenze richieste

	1. Nel tuo repository locale, vai alla directory: `fsx_deployment`
	`cd fsx_deployment`
	2. Eseguire il comando :
	`terraform destroy -var-file terraform.tfvars`	

risoluzione dei problemi

Problema	Soluzione
FSx for Lustre restituisce errori.	Per assistenza sui problemi FSx relativi a Lustre, consulta Risoluzione dei problemi di Amazon FSx for Lustre nella documentazione di FSx for Lustre.

Risorse correlate

- Creazione di Amazon FSx for Lustre utilizzando Terraform (riferimento al AWS provider nella documentazione di Terraform)
- Guida introduttiva ad Amazon FSx for Lustre (documentazione FSx per Lustre)
- AWS post di blog su Amazon FSx for Lustre

Configura una dashboard di monitoraggio Grafana per AWS ParallelCluster

Dario La Porta e William Lu, Amazon Web Services

Riepilogo

AWS ti ParallelCluster aiuta a distribuire e gestire cluster HPC (High Performance Computing). Supporta gli strumenti di pianificazione dei lavori open source AWS Batch e Slurm. Sebbene AWS ParallelCluster sia integrato con Amazon CloudWatch per la registrazione e le metriche, non fornisce una dashboard di monitoraggio per il carico di lavoro.

La dashboard Grafana per AWS ParallelCluster (GitHub) è una dashboard di monitoraggio per AWS. ParallelCluster Fornisce informazioni dettagliate sulla pianificazione dei lavori e metriche di monitoraggio dettagliate a livello di sistema operativo (OS). Per ulteriori informazioni sui dashboard inclusi in questa soluzione, consulta Dashboard di esempio nel repository. GitHub Queste metriche consentono di comprendere meglio il carico di lavoro HPC e le relative prestazioni. Tuttavia, il codice del dashboard non viene aggiornato per le versioni più recenti di AWS ParallelCluster o per i pacchetti open source utilizzati nella soluzione. Questo modello migliora la soluzione per offrire i seguenti vantaggi:

- Supporta AWS ParallelCluster v3
- Utilizza l'ultima versione dei pacchetti open source, tra cui Prometheus, Grafana, Prometheus Slurm Exporter e NVIDIA DCGM-Exporter
- Aumenta il GPUs numero di core della CPU e quelli utilizzati dai job Slurm
- Aggiunge una dashboard di monitoraggio dei lavori
- Migliora la dashboard di monitoraggio dei nodi GPU per i nodi con 4 o 8 unità di elaborazione grafica () GPUs

Questa versione della soluzione avanzata è stata implementata e verificata nell'ambiente di produzione HPC di un cliente AWS.

Prerequisiti e limitazioni

Prerequisiti

- AWS ParallelCluster CLI, installata e configurata.

- Una configurazione di rete supportata per AWS ParallelCluster. Questo modello utilizza AWS ParallelCluster utilizzando una configurazione a due sottoreti, che richiede una sottorete pubblica, una sottorete privata, un gateway Internet e un gateway NAT.
- Tutti i nodi ParallelCluster del cluster AWS devono avere accesso a Internet. Ciò è necessario affinché gli script di installazione possano scaricare il software open source e le immagini Docker.
- Una coppia di chiavi in Amazon Elastic Compute Cloud (Amazon EC2). Le risorse che hanno questa coppia di key pair hanno accesso Secure Shell (SSH) al nodo principale.

Limitazioni

- Questo pattern è progettato per supportare Ubuntu 20.04 LTS. Se utilizzi una versione diversa di Ubuntu o se usi Amazon Linux o CentOS, devi modificare gli script forniti con questa soluzione. Queste modifiche non sono incluse in questo schema.

Versioni del prodotto

- Ubuntu 20.04 LTS
- ParallelCluster 3.X

Considerazioni sulla fatturazione e sui costi

- La soluzione implementata secondo questo schema non è coperta dal livello gratuito. Si applicano costi per Amazon EC2, Amazon FSx for Lustre, il gateway NAT in Amazon VPC e Amazon Route 53.

Architecture

Architettura Target

Il diagramma seguente mostra come un utente può accedere alla dashboard di monitoraggio per AWS ParallelCluster sul nodo principale. Il nodo principale esegue NICE DCV, Prometheus, Grafana, Prometheus Slurm Exporter, Prometheus Node Exporter e NGINX Open Source. I nodi di calcolo eseguono Prometheus Node Exporter e eseguono anche NVIDIA DCGM-Exporter se il nodo contiene. GPUs Il nodo principale recupera le informazioni dai nodi di calcolo e visualizza tali dati nella dashboard di Grafana.

Nella maggior parte dei casi, il nodo principale non è sovraccaricato perché il job scheduler non richiede una quantità significativa di CPU o memoria. Gli utenti accedono alla dashboard sul nodo principale utilizzando SSL sulla porta 443.

Tutti gli spettatori autorizzati possono visualizzare in modo anonimo le dashboard di monitoraggio. Solo l'amministratore Grafana può modificare i dashboard. Si configura una password per l'amministratore Grafana nel `aws-parallelcluster-monitoring/docker-compose/docker-compose.head.yml` file.

Tools (Strumenti)

Servizi AWS

- NICE DCV è un protocollo di visualizzazione remota ad alte prestazioni che consente di fornire desktop remoti e lo streaming di applicazioni da qualsiasi cloud o data center a qualsiasi dispositivo, in condizioni di rete variabili.
- AWS ti ParallelCluster aiuta a distribuire e gestire cluster HPC (High Performance Computing). Supporta gli strumenti di pianificazione dei lavori open source AWS Batch e Slurm.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito.

Altri strumenti

- Docker è un insieme di prodotti Platform as a Service (PaaS) che utilizzano la virtualizzazione a livello di sistema operativo per fornire software in container.
- Grafana è un software open source che ti aiuta a interrogare, visualizzare, avvisare ed esplorare metriche, log e tracce.
- NGINX Open Source è un server web open source e un reverse proxy.
- NVIDIA Data Center GPU Manager (DCGM) è una suite di strumenti per la gestione e il monitoraggio delle unità di elaborazione grafica dei data center NVIDIA () in ambienti cluster. GPUs In questo modello, si utilizza DCGM-Exporter, che consente di esportare le metriche della GPU da Prometheus.
- Prometheus è un toolkit di monitoraggio del sistema open source che raccoglie e archivia le sue metriche come dati di serie temporali con coppie chiave-valore associate, chiamate etichette. In

questo modello, si utilizza anche Prometheus Slurm Exporter per raccogliere ed esportare metriche e si utilizza Prometheus Node Exporter per esportare le metriche dai nodi di calcolo.

- Ubuntu è un sistema operativo open source basato su Linux progettato per server aziendali, desktop, ambienti cloud e IoT.

Archivio di codici

Il codice per questo pattern è disponibile nel GitHub pcluster-monitoring-dashboardrepository.

Epiche

Crea le risorse necessarie

Operazione	Description	Competenze richieste
Crea un bucket S3.	Creare un bucket Amazon S3. Questo bucket viene utilizzato per archiviare gli script di configurazione. Per istruzioni, consulta Creazione di un bucket nella documentazione di Amazon S3.	Informazioni generali su AWS
Clonare il repository.	Clona il GitHub pcluster-monitoring-dashboardrepository eseguendo il seguente comando. `git clone https://github.com/aws-samples/parallelcluster-monitoring-dashboard.git`	DevOps ingegnere
Crea una password di amministratore.	1. Scegli la `aws-parallelcluster-monitoring` cartella, scegli la `docker-compose` cartella	Scripting con Linux Shell

Operazione	Description	Competenze richieste
	e quindi apri il file docker-compose.head.yml. 2. Nella `GF_SECURITY_ADMIN_PASSWORD` variabile, sostituiscila con una password a tua scelta. `Grafana4PC!` Questa è la password amministrativa che usi per gestire l'account Grafana. 3. Salva e chiudi il file docker-compose.head.yml.	
Copia i file richiesti nel bucket S3.	Copia lo script post_install.sh e la aws-parallelcluster-monitoringcartella nel bucket S3 che hai creato. Per istruzioni, consulta Caricamento di oggetti nella documentazione di Amazon S3.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Configura un gruppo di sicurezza aggiuntivo per il nodo principale.	1. Crea un gruppo di sicurezza per il nodo principale. Questo gruppo di sicurezza consentirà il traffico in entrata verso i dashboard di monitoraggio sul nodo principale. Per istruzioni, consulta Creare un gruppo di sicurezza nella documentazione di Amazon VPC. 2. Aggiungi una regola in entrata al gruppo di sicurezza. Per istruzioni, consulta Aggiungere regole a un gruppo di sicurezza nella documentazione di Amazon VPC. Utilizza i seguenti parametri per la regola: • Tipo: HTTPS • Protocollo: TCP • Intervallo di porte: 443 • Fonte: inserisci il tuo indirizzo IP • Descrizione: consente agli utenti di accedere alla dashboard di monitoraggio	Amministratore AWS

Operazione	Description	Competenze richieste
Configura una policy IAM per il nodo principale.	Crea una policy basata sull'identità per il nodo principale. Questa policy consente al nodo di recuperare i dati metrici da Amazon. CloudWatch Il GitHub repository contiene una policy di esempio. Per istruzioni, consulta Creazione di policy IAM nella documentazione di AWS Identity and Access Management (IAM).	Amministratore AWS
Configura una policy IAM per i nodi di calcolo.	Crea una policy basata sull'identità per i nodi di calcolo. Questa politica consente al nodo di creare i tag che contengono l'ID del lavoro e il proprietario del lavoro. Il GitHub repository contiene un esempio di policy. Per istruzioni, consulta Creazione di politiche IAM nella documentazione IAM. Se utilizzi il file di esempio fornito, sostituisci i seguenti valori: • <REGION>— La regione AWS in cui è ospitato il cluster • <ACCOUNT_ID>— L'ID dell'account AWS	Amministratore AWS

Creazione del cluster

Operazione	Description	Competenze richieste
Modifica il file modello di cluster fornito.	Crea il ParallelCluster cluster AWS. Utilizza il file modello CloudFormation AWS cluster.yaml fornito come punto di partenza per creare il cluster. Sostituisci i seguenti valori nel modello fornito: • <REGION>— La regione AWS in cui è ospitato il cluster. • <HEADNODE_SUBNET>— La sottorete pubblica del VPC. • <ADDITIONAL_HEAD_NODE_SG>— Il nome del gruppo di sicurezza creato per il nodo principale. • <KEY_NAME>— Inserisci il nome di una coppia di EC2 chiavi Amazon esistente. Le risorse che hanno questa coppia di key pair hanno accesso Secure Shell (SSH) al nodo principale. • <ALLOWED_IPS>-—Immettere l'intervallo di indirizzi IP in formato CIDR a cui è consentito effettuare connessioni SSH al nodo principale.	Amministratore AWS

Operazione	Description	Competenze richieste
	• <ADDITIONAL_HEAD_NODE_POLICY>— Inserisci il nome della policy IAM che hai creato per il nodo principale. • <BUCKET_NAME>— Inserisci il nome del bucket S3 che hai creato. • <COMPUTE_SUBNET>— Inserisci il nome della sottorete privata nel VPC. • <ADDITIONAL_COMPUTE_NODE_POLICY>— Inserisci il nome della policy IAM che hai creato per il nodo di calcolo.	
Crea il cluster.	Nella ParallelCluster CLI di AWS, inserisci il seguente comando. Questo distribuisce il CloudFormation modello e crea il cluster. Per ulteriori informazioni su questo comando, consulta pcluster create-cluster nella documentazione AWS. ParallelCluster `pcluster create-cluster -n <cluster_name> -c cluster.yaml`	Amministratore AWS

Operazione	Description	Competenze richieste
Monitora la creazione del cluster.	Immettere il seguente comando per monitorare la creazione del cluster. Per ulteriori informazioni su questo comando, consulta pcluster describe-cluster nella documentazione AWS. ParallelCluster `pcluster describe-cluster -n <cluster_name>`	Amministratore AWS

Utilizzo delle dashboard Grafana

Operazione	Description	Competenze richieste
Accesso al portale Grafana.	1. Immettere il seguente comando per recuperare l'indirizzo IP pubblico del nodo principale. `pcluster describe-cluster -n <cluster_name> --query headNode.publicIpAddress` 2. In un browser Web, vai al seguente URL per accedere alla dashboard di Grafana. https://<head_node_public_ip_address>	Amministratore AWS

Operazione	Description	Competenze richieste
	3. Nella prima pagina di Grafana, scegli l'icona quadrata della Dashboard nel menu a sinistra, quindi scegli Generale. Questo mostra un elenco di dashboard configurati. Le seguenti dashboard sono disponibili in Grafana: • Costo del cluster: contiene informazioni sul costo del cluster • Registri del cluster: contiene informazioni sui log del cluster • Dettagli sui nodi di calcolo: contiene informazioni sulle statistiche di utilizzo dei nodi di calcolo • Elenco dei nodi di calcolo: contiene l'elenco dei nodi di calcolo del cluster • Nodi GPU: contiene informazioni sulle statistiche di utilizzo dei nodi GPU • Dettagli sui lavori: contiene informazioni sull'utilizzo delle risorse relative ai lavori	

Operazione	Description	Competenze richieste
	• Dettagli del nodo principale: contiene informazioni sulle statistiche di utilizzo del nodo principale • ParallelCluster Riepilogo: contiene informazioni sull'utilizzo del cluster	

Pulisci la soluzione per evitare di incorrere nei costi associati

Operazione	Description	Competenze richieste
Elimina il cluster.	Immettere il seguente comando per eliminare il cluster. Per ulteriori informazioni su questo comando, consulta pcluster delete-cluster nella documentazione AWS. ParallelCluster `pcluster delete-cluster -n <cluster_name>`	Amministratore AWS
Elimina le politiche IAM.	Elimina le policy che hai creato per il nodo principale e il nodo di calcolo. Per ulteriori informazioni sull'eliminazione delle policy, consulta Eliminazione delle policy IAM nella documentazione IAM.	Amministratore AWS
Elimina il gruppo e la regola di sicurezza.	Eliminare il gruppo di sicurezza creato per il nodo principale. Per ulteriori	Amministratore AWS

Operazione	Description	Competenze richieste
	informazioni, consulta Eliminare le regole del gruppo di sicurezza ed Eliminare un gruppo di sicurezza nella documentazione di Amazon VPC.	
Eliminare il bucket S3.	Elimina il bucket S3 che hai creato per archiviare gli script di configurazione. Per ulteriori informazioni, consulta Eliminazione di un bucket nella documentazione di Amazon S3.	Informazioni generali su AWS

risoluzione dei problemi

Problema	Soluzione
Il nodo principale non è accessibile nel browser.	Controlla il gruppo di sicurezza e conferma che la porta in ingresso 443 sia aperta.
Grafana non si apre.	Sul nodo principale, controlla il registro del contenitore per`docker logs Grafana`.
Alcune metriche non contengono dati.	Sul nodo principale, controlla i log dei contenito ri di tutti i contenitori.

Risorse correlate

Documentazione AWS

- Politiche IAM per Amazon EC2

Altre risorse AWS

- AWS ParallelCluster
- Dashboard di monitoraggio per AWS ParallelCluster (post sul blog AWS)

Altre risorse

- Sistema di monitoraggio Prometheus
- Grafana

Altri modelli

- Implementa la diagnostica e la risoluzione dei problemi di Kubernetes basate sull'intelligenza artificiale con K8SGPT e l'integrazione con Amazon Bedrock

Cloud ibrido

Argomenti

- Configura una CI/CD pipeline per carichi di lavoro ibridi su Amazon ECS Anywhere utilizzando AWS CDK e GitLab
- Altri modelli

Configura una CI/CD pipeline per carichi di lavoro ibridi su Amazon ECS Anywhere utilizzando AWS CDK e GitLab

Rafael Ortiz, Amazon Web Services

Riepilogo

Amazon ECS Anywhere è un'estensione di Amazon Elastic Container Service (Amazon ECS). Fornisce supporto per la registrazione di un'istanza esterna, come un server locale o una macchina virtuale (VM), nel cluster Amazon ECS. Questa funzionalità aiuta a ridurre i costi e mitigare l'orchestrazione e le operazioni complesse dei container locali. Puoi utilizzare ECS Anywhere per distribuire ed eseguire applicazioni container in ambienti locali e cloud. Elimina la necessità per il team di apprendere più domini e set di competenze o di gestire software complessi da solo.

Questo modello descrive un step-by-step approccio per il provisioning di un cluster Amazon ECS con istanze Amazon ECS Anywhere utilizzando gli stack Amazon Web Services (AWS) Cloud Development Kit (AWS CDK). Quindi usi AWS CodePipeline per configurare una pipeline di integrazione e distribuzione continua (CI/CD). Quindi, replichi il tuo repository di GitLab codice su AWS CodeCommit e distribuisci la tua applicazione containerizzata sul cluster Amazon ECS.

Questo modello è progettato per aiutare coloro che utilizzano l'infrastruttura locale a eseguire applicazioni container e a gestire il codice base dell'applicazione GitLab . Puoi gestire questi carichi di lavoro utilizzando i servizi cloud AWS, senza disturbare l'infrastruttura locale esistente.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un'applicazione contenitore in esecuzione su un'infrastruttura locale.
- Un GitLab repository in cui gestire la base di codice dell'applicazione. Per ulteriori informazioni, vedete Repository ()GitLab.
- AWS Command Line Interface (AWS CLI), installata e configurata. Per ulteriori informazioni, consulta Installazione o aggiornamento della versione più recente dell'interfaccia a riga di comando di AWS (documentazione dell'interfaccia a riga di comando di AWS).
- AWS CDK Toolkit, installato e configurato a livello globale. Per ulteriori informazioni, consulta Installare il CDK AWS (documentazione AWS CDK).

- npm, installato e configurato per AWS CDK in. TypeScript Per ulteriori informazioni, consulta Download e installazione di Node.js e npm (documentazione npm).

Limitazioni

- Per limitazioni e considerazioni, consulta Istanze esterne (Amazon ECS Anywhere) nella documentazione di Amazon ECS.

Versioni del prodotto

- AWS CDK Toolkit versione 2.27.0 o successiva
- npm versione 7.20.3 o successiva
- Node.js versione 16.6.1 o successiva

Architecture

Stack tecnologico Target

- AWS CDK
- AWS CloudFormation
- AWS CodeBuild
- AWS CodeCommit
- AWS CodePipeline
- Amazon ECS Anywhere
- Amazon Elastic Container Registry (Amazon ECR)
- AWS Identity and Access Management (IAM)
- Gestore di sistema AWS
- GitLab deposito

Architettura Target

Questo diagramma rappresenta due flussi di lavoro principali descritti in questo modello, il provisioning del cluster Amazon ECS e la configurazione della CI/CD pipeline che configura e distribuisce la pipeline, come segue: CI/CD

1. Eseguire il provisioning del cluster Amazon ECS
   a. Quando distribuisci il primo stack CDK AWS, viene creato uno CloudFormation stack su AWS.
   b. Questo CloudFormation stack fornisce un cluster Amazon ECS e le relative risorse AWS.
   c. Per registrare un'istanza esterna con un cluster Amazon ECS, devi installare AWS Systems Manager Agent (SSM Agent) sulla tua macchina virtuale e registrare la macchina virtuale come istanza gestita da AWS Systems Manager.
   d. È inoltre necessario installare l'agente contenitore Amazon ECS e Docker sulla macchina virtuale per registrarla come istanza esterna nel cluster Amazon ECS.
   e. Quando l'istanza esterna è registrata e configurata con il cluster Amazon ECS, può eseguire più contenitori sulla macchina virtuale, che è registrata come istanza esterna.
   f. Il cluster Amazon ECS è attivo e può eseguire i carichi di lavoro delle applicazioni tramite contenitori. L'istanza del contenitore Amazon ECS Anywhere viene eseguita in un ambiente locale ma è associata al cluster Amazon ECS nel cloud.
2. Configurazione e distribuzione della pipeline CI/CD
   a. Quando distribuisci il secondo stack CDK AWS, viene creato un altro CloudFormation stack su AWS.
   b. Questo CloudFormation stack fornisce una pipeline CodePipeline e le relative risorse AWS.
   c. Invia e unisci le modifiche al codice dell'applicazione in un repository locale. GitLab
   d. Il GitLab repository viene replicato automaticamente nel repository. CodeCommit
   e. Gli aggiornamenti al repository vengono avviati automaticamente CodeCommit . CodePipeline
   f. CodePipeline copia il codice CodeCommit e crea l'applicazione incorporata distribuibile. CodeBuild
   g. CodePipeline crea un'immagine Docker dell'ambiente di CodeBuild compilazione e la invia al repository Amazon ECR.
   h. CodePipeline avvia CodeDeploy azioni che estraggono l'immagine del contenitore dal repository Amazon ECR.
   i. CodePipeline distribuisce l'immagine del contenitore sul cluster Amazon ECS.

Automazione e scalabilità

Questo modello utilizza AWS CDK come strumento Infrastructure as Code (IaC) per configurare e distribuire questa architettura. AWS CDK ti aiuta a orchestrare le risorse AWS e configurare Amazon ECS Anywhere e la pipeline. CI/CD

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software che aiuta a definire e fornire l'infrastruttura cloud AWS in codice.
- AWS CodeCommit è un servizio di controllo delle versioni che consente di archiviare e gestire archivi Git in modo privato, senza dover gestire il proprio sistema di controllo del codice sorgente.
- AWS ti CodePipeline aiuta a modellare e configurare rapidamente le diverse fasi di un rilascio di software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.
- Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di immagini di container gestito sicuro, scalabile e affidabile.
- Amazon Elastic Container Service (Amazon ECS) è un servizio rapido e scalabile di gestione dei container che ti aiuta a eseguire, arrestare e gestire container in un cluster. Questo modello utilizza anche Amazon ECS Anywhere, che fornisce supporto per la registrazione di un server o una macchina virtuale locale nel cluster Amazon ECS.

Altri strumenti

- Node.js è un ambiente di JavaScript runtime basato sugli eventi progettato per la creazione di applicazioni di rete scalabili.
- npm è un registro software che viene eseguito in un ambiente Node.js e viene utilizzato per condividere o prendere in prestito pacchetti e gestire la distribuzione di pacchetti privati.
- Vagrant è un'utilità open source per la creazione e la manutenzione di ambienti di sviluppo software virtuali portatili. A scopo dimostrativo, questo modello utilizza Vagrant per creare una macchina virtuale locale.

Archivio di codice

Il codice per questo pattern è disponibile nella pipeline GitHub CI/CD per Amazon ECS Anywhere utilizzando il repository AWS CDK.

Best practice

Prendi in considerazione le seguenti best practice per la distribuzione di questo pattern:

- Le migliori pratiche per lo sviluppo e la distribuzione di infrastrutture cloud con AWS CDK
- Le migliori pratiche per lo sviluppo di applicazioni cloud con AWS CDK (post sul blog AWS)

Epiche

Verifica la configurazione di AWS CDK

Operazione	Description	Competenze richieste
Verifica la versione di AWS CDK.	Verifica la versione di AWS CDK Toolkit inserendo il seguente comando. `cdk --version` Questo modello richiede la versione 2.27.0 o successiva. Se disponi di una versione precedente, segui le istruzioni nella documentazione di AWS CDK per aggiornarla.	DevOps ingegnere
Verifica la versione di npm.	Verifica la versione di npm inserendo il seguente comando. `npm --version` Questo modello richiede la versione 7.20.3 o successiva. Se hai una versione precedente, segui le istruzioni nella	DevOps ingegnere

Operazione	Description	Competenze richieste
	documentazione di npm per aggiornarla.	

Configura le credenziali AWS.	Configura le credenziali AWS inserendo il `aws configure` comando e seguendo le istruzioni.
	
	`$aws configure`
	`AWS Access Key ID [None]: <your-access-key-ID>`
	`AWS Secret Access Key [None]: <your-secret-access-key>`
	`Default region name [None]: <your-Region-name>`
	`Default output format [None]:`
		DevOps ingegnere

Avvia l'ambiente CDK AWS

Operazione	Description	Competenze richieste
Clona il repository di codice AWS CDK.	1. Clona la pipeline CI/CD per Amazon ECS Anywhere utilizzando il repository AWS CDK per questo pattern inserendo il seguente comando. `git clone https://github.com/aws-samples/amazon-ecs-any`	DevOps ingegnere

Operazione	Description	Competenze richieste

	`where-cicd-pipelin e-cdk-sample.git`
	2. Naviga nella directory clonata inserendo il seguente comando.
	`cd amazon-ecs-anywher e-cicd-pipeline-cd k-sample`	

Avvia l'ambiente.	Distribuisci il CloudForm ation modello nell'account e nella regione AWS che desideri utilizzare inserendo il seguente comando. `cdk bootstrap <account-number>/<Region>` Per ulteriori informazioni, consulta Bootstrapping nella documentazione di AWS CDK.	DevOps ingegnere

Crea e distribuisci l'infrastruttura per Amazon ECS Anywhere

Operazione	Description	Competenze richieste

Installa le dipendenze del pacchetto e compila i TypeScript file.	Installa le dipendenze del pacchetto e compila TypeScrip t i file inserendo i seguenti comandi.
	`$cd EcsAnywhereCdk`
	`$npm install`
		DevOps ingegnere

Operazione	Description	Competenze richieste
	`$npm fund` Questi comandi installano tutti i pacchetti dal repository di esempio. Per ulteriori informazioni, consulta npm ci e npm install nella documentazione di npm. Se riscontri errori sui pacchetti mancanti quando inserisci questi comandi, consulta la sezione Risoluzione dei problemi di questo modello.	
Compilare il progetto.	Per creare il codice del progetto, inserisci il seguente comando. `npm run build` Per ulteriori informazioni sulla creazione e la distribuzione del progetto, consulta La tua prima app AWS CDK nella documentazione di AWS CDK.	DevOps ingegnere

Operazione	Description	Competenze richieste

Implementa lo stack di infrastruttura Amazon ECS Anywhere.	1. Elenca gli stack inserendo il seguente comando.
	`$cdk list`
	2. Verifica che l'output restituisca le `EcsAnywhereInfraStack` `ECSAnywherePipelineStack` pile e.
	3. Distribuisci lo `EcsAnywhereInfraStack` stack inserendo il seguente comando.
	`$cdk deploy EcsAnywhereInfraStack`
		DevOps ingegnere

Operazione	Description	Competenze richieste
Verifica la creazione e l'output dello stack.	1. Accedi alla Console di gestione AWS e apri la CloudFormation console all'indirizzo https://console.aws.amazon.com/cloudformation/. 2. Nella pagina Stacks, seleziona lo `EcsAnywhereInfraStack` stack. 3. Verifica che lo stato dello stack sia o. `CREATE_IN_PROGRESS` `CREATE_COMPLETE` La configurazione del cluster Amazon ECS può richiedere del tempo. Non procedere fino al completamento della creazione dello stack.	DevOps ingegnere

Configura una macchina virtuale locale

Operazione	Description	Competenze richieste
Configura la tua VM.	Crea una VM Vagrant inserendo il `vagrant up` comando dalla directory principale in cui si trova Vagrantfile. Per ulteriori informazioni, consulta la documentazione di Vagrant.	DevOps ingegnere

Operazione	Description	Competenze richieste

Registra la tua macchina virtuale come istanza esterna.	1. Accedi alla VM Vagrant utilizzando il comando. `vagrant ssh` Per ulteriori informazioni, consulta la documentazione di Vagrant.
	2. Installa AWS CLI sulla macchina virtuale seguendo le istruzioni di installazione dell'interfaccia a riga di comando di AWS e inserendo i seguenti comandi.
	
	`$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" \`
	`> -o "awscliv2.zip"`
	`$sudo apt install unzip`
	`$unzip awscliv2.zip`
	`$sudo ./aws/install`
	`$aws configure`
	`AWS Access Key ID [None]: <your-access-key-ID>`
	`AWS Secret Access Key [None]: <your-secret-access-key>`
	`Default region name [None]: <your-Region-name>`
	`Default output format [None]:`
	
	1. Crea un codice di attivazione e un ID che puoi utilizzar
		DevOps ingegnere

Operazione	Description	Competenze richieste

	e per registrare la tua macchina virtuale con AWS Systems Manager e attivare l'istanza esterna. L'output di questo comando include l'ID di attivazione e i valori del codice di attivazione.
	
	`aws ssm create-activation \ > --iam-role EcsAnywhereInstanceRole \ > | tee ssm-activation.json`
	
	Se ricevi un errore quando esegui questo comando, consulta la sezione Risoluzione dei problemi.
	2. Esporta l'ID di attivazione e i valori del codice.
	
	`export ACTIVATION_ID=<activation-ID> export ACTIVATION_CODE=<activation-code>`
	3. Scarica lo script di installazione sulla tua macchina virtuale.
	
	`curl --proto "https" -o "ecs-anywhere-install.sh" \`	

Operazione	Description	Competenze richieste

	`> "https://amazon-ecs-agent.s3.amazonaws.com/ecs-anywhere-install-latest.sh"`
	4. Esegui lo script di installazione sulla tua macchina virtuale.
	`sudo bash ecs-anywhere-install.sh \ --cluster EcsAnywhereCluster \ --activation-id $ACTIVATION_ID \ --activation-code $ACTIVATION_CODE \ --region <region-name>`
	Questo configura la tua macchina virtuale come un'istanza esterna di Amazon ECS Anywhere e registra l'istanza nel cluster Amazon ECS. Per ulteriori informazioni, consulta Registrazione di un'istanza esterna in un cluster nella documentazione di Amazon ECS. In caso di problemi, consulta la sezione Risoluzione dei problemi.	

Operazione	Description	Competenze richieste

Verifica lo stato di Amazon ECS Anywhere e della macchina virtuale esterna.	Per verificare se la tua macchina virtuale è connessa al piano di controllo di Amazon ECS e se è in esecuzione, usa i seguenti comandi.
	`$aws ssm describe-instance-information`
	`$aws ecs list-container-instances --cluster $CLUSTER_NAME`
		DevOps ingegnere

Implementa la pipeline CI/CD

Operazione	Description	Competenze richieste
Crea un ramo nel CodeCommit repository.	Crea un ramo denominato `main` nel CodeCommit repository creando il primo commit per il repository. Puoi seguire la documentazione di AWS per creare un commit in CodeCommit. Il comando seguente è un esempio. `aws codecommit put-file \ --repository-name EcsAnywhereRepo \ --branch-name main \ --file-path README.md \ --file-content "Test" \ --name "Dev Ops" \`	DevOps ingegnere

Operazione	Description	Competenze richieste
	`--email "devops@example.com" \ --commit-message "Adding README."`	
Configura il mirroring dei repository.	È possibile eseguire il mirroring di un GitLab repository da e verso fonti esterne. È possibile selezionare quale repository funge da origine. I rami, i tag e i commit vengono sincronizzati automaticamente. Configura un push mirror tra il GitLab repository che ospita l'applicazione e il repository. CodeCommit Per istruzioni, consultate Configurare un push mirror da GitLab a CodeCommit (GitLab documentazione). **Note** Per impostazione predefinita, il mirroring sincronizza automaticamente il repository. Se desideri aggiornare manualmente i repository, consulta Aggiornare un mirror (documentazione). GitLab	DevOps ingegnere

Operazione	Description	Competenze richieste
Implementa lo stack della CI/CD pipeline.	Distribuisci lo `EcsAnywhe rePipelineStack` stack inserendo il seguente comando. `$cdk  deploy EcsAnywhe rePipelineStack`	DevOps ingegnere

Operazione	Description	Competenze richieste
Testa la CI/CD pipeline.	1. Apporta modifiche al codice dell'applicazione e invialo al repository di origine locale GitLab . Ad esempio, modificate il `../application/index.html` file per aggiornare il valore della versione dell'applicazione. 2. Quando il codice viene replicato nel CodeCommit repository, viene avviata la CI/CD pipeline. Esegui una delle seguenti operazioni: • Se utilizzi il mirroring automatico per sincronizzare il repository con il GitLab repository, continua con il CodeCommit passaggio successivo. • Se utilizzi il mirroring manuale, invia le modifiche al codice dell'applicazione al CodeCommit repository seguendo le istruzioni in Aggiornare un mirror (documentazione). GitLab 3. Sul computer locale, in un browser Web, immettete http://localhost:80. Questo apre la pagina web di NGINX perché la porta 80	DevOps ingegnere

Operazione	Description	Competenze richieste
	viene inoltrata a localhost in Vagrantfile. Conferma di poter visualizzare il valore della versione aggiornat a dell'applicazione. Ciò convalida la distribuzione della pipeline e dell'imma gine. 4. (Facoltativo) Se desideri verificare la distribuzione nella Console di gestione AWS, procedi come segue: a. Aprire la console Amazon ECS in https://console.aws.amazon.com/ecs/. b. Seleziona la Regione da utilizzare nella barra di navigazione. c. Nel pannello di navigazio ne scegli Cluster. d. Nella pagina Cluster seleziona il cluster EcsAnywhereCluster. e. Scegli Task Definitions. f. Conferma che il contenitore è in funzione.	

Eliminazione

Operazione	Description	Competenze richieste

Pulisci ed elimina le risorse.	Dopo aver seguito questo schema, dovresti rimuovere le proof-of-concept risorse che hai creato. Per pulire, inserisci i seguenti comandi.
	`$cdk destroy EcsAnywherePipelineStack`
	`$cdk destroy EcsAnywhereInfraStack`
		DevOps ingegnere

Risoluzione dei problemi

Problema	Soluzione

Errori relativi ai pacchetti mancanti durante l'installazione delle dipendenze dei pacchetti.	Immettete uno dei seguenti comandi per risolvere i pacchetti mancanti.
	`$npm ci`
	or
	`$npm install -g @aws-cdk/<package_name>`

Quando si esegue il `aws ssm create-activation` comando sulla macchina virtuale, viene visualizzato il seguente errore. `An error occurred (ValidationException) when calling the CreateActivation operation: Nonexistent role or missing ssm`	Lo `EcsAnywhereInfraStack` stack non è completamente distribuito e il ruolo IAM necessario per eseguire questo comando non è stato ancora creato. Controlla lo stato dello stack nella console. CloudFormation Riprova il comando dopo che lo stato è cambiato in. `CREATE_COMPLETE`

Problema	Soluzione
`service principal in trust policy: arn:aws:iam::000000000000:role/EcsAnywhereInstanceRole`	

Viene `UNHEALTHY` restituito un controllo dello stato di Amazon ECS e viene visualizzato il seguente errore nella sezione Servizi del cluster nella console Amazon ECS.
 	
`service EcsAnywhereService was unable to place a task because no container instance met all of its requirements. Reason: No Container Instances were found in your cluster.`
	Riavvia l'agente Amazon ECS sulla tua macchina virtuale Vagrant inserendo i seguenti comandi.
	
	`$vagrant ssh`
	`$sudo systemctl restart ecs`
	`$sudo systemctl status ecs`

Risorse correlate

- Pagina di marketing di Amazon ECS Anywhere
- Documentazione di Amazon ECS Anywhere
- Dimostrazione di Amazon ECS Anywhere (video)
- Esempi GitHub di workshop Amazon ECS Anywhere ()
- Mirroring del repository (documentazione) GitLab

Altri modelli

- Automatizza la configurazione del peering interregionale con AWS Transit Gateway
- Distribuisci applicazioni containerizzate in AWS IoT Greengrass V2 esecuzione come contenitore Docker
- Gestisci le applicazioni container locali configurando Amazon ECS Anywhere con AWS CDK
- Modifica le intestazioni HTTP durante la migrazione da F5 a un Application Load Balancer su AWS
- Rehosting dei carichi di lavoro locali nel cloud AWS: lista di controllo per la migrazione
- Usa le query BMC Discovery per estrarre i dati di migrazione per la pianificazione della migrazione

Gestione e governance

Argomenti

- Identifica e avvisa quando le risorse Amazon Data Firehose non sono crittografate con una chiave AWS KMS
- Automatizza le allocazioni IPv4 IPAM CIDR di Amazon VPC per nuove applicazioni utilizzando AFT Account AWS
- Automatizza l'aggiunta o l'aggiornamento delle voci di registro di Windows utilizzando AWS Systems Manager
- Crea automaticamente un RFC in AMS usando Python
- Arresta e avvia automaticamente un'istanza database Amazon RDS utilizzando AWS Systems Manager Maintenance Windows
- Centralizza la distribuzione dei pacchetti software in AWS Organizations utilizzando Terraform
- Configura la registrazione per le applicazioni.NET in Amazon CloudWatch Logs utilizzando NLog
- Copia i prodotti AWS Service Catalog su diversi account AWS e regioni AWS
- Crea una matrice RACI o RASCI per un modello operativo cloud
- Crea allarmi per metriche personalizzate utilizzando il rilevamento delle anomalie di Amazon CloudWatch
- Crea un IDE AWS Cloud9 che utilizza volumi Amazon EBS con crittografia predefinita
- Crea CloudWatch dashboard Amazon basate su tag automaticamente
- Documenta il progetto della tua landing zone AWS
- Migliora le prestazioni operative abilitando Amazon DevOps Guru su più regioni AWS, account e OUs con AWS CDK
- Gestisci i set di autorizzazioni per più account utilizzando Account Factory for Terraform
- Implementa Account Factory for Terraform (AFT) utilizzando una pipeline bootstrap
- Gestisci i prodotti AWS Service Catalog in più account AWS e regioni AWS
- Monitora i cluster SAP RHEL Pacemaker utilizzando i servizi AWS
- Monitora l'attività delle applicazioni utilizzando CloudWatch Logs Insights
- Monitora l'uso di un'Amazon Machine Image condivisa su più Account AWS
- Visualizza i dettagli degli snapshot EBS per il tuo account o la tua organizzazione AWS
- Altri modelli

Identifica e avvisa quando le risorse Amazon Data Firehose non sono crittografate con una chiave AWS KMS

Ram Kandaswamy, Amazon Web Services

Riepilogo

Per motivi di conformità, alcune organizzazioni devono avere la crittografia abilitata su risorse di distribuzione dei dati come Amazon Data Firehose. Questo modello mostra un modo per monitorare, rilevare e notificare quando le risorse non sono conformi.

Per mantenere i requisiti di crittografia, questo modello può essere utilizzato AWS per fornire il monitoraggio e il rilevamento automatici delle risorse di distribuzione di Amazon Data Firehose che non sono crittografate con una chiave AWS Key Management Service (AWS KMS). La soluzione invia notifiche di avviso e può essere estesa per eseguire riparazioni automatiche. Questa soluzione può essere applicata a un account singolo o a un ambiente con più account, ad esempio un ambiente che utilizza una AWS landing zone o. AWS Control Tower

Prerequisiti e limitazioni

Prerequisiti

- Flusso di distribuzione di Amazon Data Firehose
- Autorizzazioni e familiarità sufficienti con CloudFormation, che vengono utilizzate in questa automazione dell'infrastruttura

Limitazioni

- La soluzione non è in tempo reale perché utilizza AWS CloudTrail gli eventi per il rilevamento e c'è un ritardo tra il momento in cui viene creata una risorsa non crittografata e l'invio della notifica.

Architecture

Stack tecnologico Target

La soluzione utilizza la tecnologia serverless e i seguenti servizi:

- AWS CloudTrail
- Amazon CloudWatch

- AWS Command Line Interface (AWS CLI)
- AWS Identity and Access Management (IAM)
- Amazon Data Firehose
- AWS Lambda
- Amazon Simple Notification Service (Amazon SNS)

Architettura Target

Il diagramma illustra questi passaggi:

1. Un utente crea o modifica Amazon Data Firehose.
2. Un CloudTrail evento viene rilevato e abbinato.
3. Lambda viene richiamata.
4. Vengono identificate le risorse non conformi.
5. Viene inviata una notifica via e-mail.

Automazione e scalabilità

È possibile utilizzare CloudFormation StackSets per applicare questa soluzione a più Regioni AWS account con un solo comando.

Tools (Strumenti)

- AWS CloudTrailè uno strumento Servizio AWS che vi aiuta ad abilitare la governance, la conformità e il controllo operativo e dei rischi del vostro Account AWS. Le azioni intraprese da un utente, da un ruolo o da un Servizio AWS vengono registrate come eventi in CloudTrail. Gli eventi includono le azioni intraprese nelle operazioni dell'API Console di gestione AWS AWS CLI AWS SDKs,, e.
- Amazon CloudWatch Events offre un flusso quasi in tempo reale di eventi di sistema che descrivono i cambiamenti nelle AWS risorse.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti consente di interagire Servizi AWS utilizzando i comandi nella shell della riga di comando.
- AWS Identity and Access Management (IAM) è un servizio web che consente di controllare in modo sicuro l'accesso alle AWS risorse. Utilizza IAM per controllare chi è autenticato (accesso effettuato) e autorizzato (dispone di autorizzazioni) per l’utilizzo di risorse.

- Amazon Data Firehose è un servizio completamente gestito per la distribuzione di dati in streaming in tempo reale. Con Firehose, non è necessario scrivere applicazioni o gestire risorse. È sufficiente configurare i produttori dati perché inviino i dati a Firehose, che li distribuirà automaticamente alla destinazione specificata.
- AWS Lambdaè un servizio di elaborazione che supporta l'esecuzione di codice senza fornire o gestire server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo. Paghi solo per il tempo di elaborazione che consumi: non è previsto alcun costo quando il codice non è in esecuzione.
- Amazon Simple Notification Service (Amazon SNS) è un servizio gestito che fornisce il recapito dei messaggi dagli editori agli abbonati (noti anche come produttori e consumatori).

Epiche

Applica la crittografia per garantire la conformità

Operazione	Description	Competenze richieste
Implementazione. CloudFormation StackSets	In AWS CLI, utilizzate il `firehose-encryption-checker.yaml` modello (allegato) per creare lo stack set eseguendo il comando seguente. Fornisci un argomento Amazon SNS valido Amazon Resource Name (ARN) per il parametro . L'implementazione dovrebbe creare correttamente le regole CloudWatch Events, la funzione Lambda e un ruolo IAM con le autorizzazioni necessarie, come descritto nel modello. `aws cloudformation create-stack-set --stack-set-name`	Architetto del cloud, amministratore di sistema

Operazione	Description	Competenze richieste
	`my-stack-set --template-body file://firehose-encryption-checker.yaml`	
Crea istanze stack.	Gli stack possono essere creati in uno o più account a tua scelta. Regioni AWS Per creare istanze stack, esegui il comando seguente. Sostituisci il nome dello stack, i numeri di account e le regioni con i tuoi. `aws cloudformation create-stack-instances --stack-set-name my-stack-set --accounts 123456789012 223456789012 --regions us-east-1 us-east-2 us-west-1 us-west-2 --operation-preferences FailureToleranceCount=1`	Architetto del cloud, amministratore di sistema

Risorse correlate

- Lavorare con CloudFormation StackSets
- Che cos'è Amazon CloudWatch Events?

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Automatizza le allocazioni IPv4 IPAM CIDR di Amazon VPC per nuove applicazioni utilizzando AFT Account AWS

Kien Pham e Alex Pazik, Amazon Web Services

Riepilogo

Questo modello mostra come automatizzare le allocazioni CIDR di Amazon VPC IP Address Manager (IPAM IPv4) per i nuovi utenti Account AWS utilizzando Account AWS Control Tower Factory for Terraform (AFT). Questa operazione viene eseguita utilizzando una personalizzazione a livello di account che alloca un blocco IPv4 CIDR da IPAM a un nuovo cloud privato virtuale (VPC) utilizzando il modulo. `aft-account-customizations`

Con IPAM, puoi organizzare, assegnare, monitorare e controllare gli indirizzi IP su larga scala, consentendoti di pianificare, tracciare e monitorare facilmente gli indirizzi IP per i tuoi carichi di lavoro. AWS È possibile creare un pool IPAM e IPAM da utilizzare per allocare un blocco IPv4 CIDR a un nuovo VPC durante il processo di vendita dell'account.

Prerequisiti e limitazioni

Prerequisiti

- Un active Account AWS with AWS Control Tower abilitato in un ambiente supportato e distribuito in AFT Regione AWS
- Un provider di sistemi di controllo della versione (VCS) supportato come BitBucket GitHub, ed Enterprise GitHub
- Interfaccia a riga di comando (CLI) Terraform installata
- Un ambiente di runtime in cui è possibile eseguire il modulo Terraform che installa AFT
- AWS Command Line Interface (AWS CLI) installato e configurato

Limitazioni

- Alcuni Servizi AWS non sono disponibili in tutti Regioni AWS. Per informazioni sulla disponibilità per regione, consulta AWS Servizi per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Versioni del prodotto

- AWS Control Tower landing zone versione 3.0 o successiva, precedente alla versione 4.0
- AFT versione 1.13.0 o successiva, precedente alla versione 2.0.0
- Terraform OSS versione 1.2.0 o successiva, precedente alla versione 2.0.0
- Terraform AWS Provider (`terraform-provider-aws`) versione 5.11.0 o successiva, precedente alla versione 6.0.0
- Modulo Terraform per IPAM () versione 2.1.0 o successiva `aws-ia/ipam/aws`

Architecture

Il diagramma seguente mostra il flusso di lavoro e i componenti di questo modello.

Il flusso di lavoro è costituito dalle seguenti attività principali:

1. Attiva modifiche: le modifiche a Terraform e alla personalizzazione IPAM vengono salvate nel GitHub repository e inviate. Questa attività attiva automaticamente la pipeline. AWS CodeBuild
2. Automatizza la compilazione: all'interno CodeBuild, vengono attivati più progetti di compilazione. AWS Step Functions
3. Applica personalizzazione: Step Functions si coordina con CodeBuild per pianificare e applicare le modifiche Terraform. Questa attività utilizza il modulo AFT Terraform per coordinare l'assegnazione dell'IP del pool IPAM all'account fornito. AWS

Tools (Strumenti)

Servizi AWS

- AWS CodeBuildè un servizio di compilazione completamente gestito che consente di compilare il codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS CodePipelineti aiuta a modellare e configurare rapidamente le diverse fasi di una versione del software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- AWS Control Towerorchestra le funzionalità di molti altri Servizi AWS, tra cui AWS Organizations, AWS Service Catalog e. AWS IAM Identity Center Può aiutarti a configurare e gestire un ambiente con AWS più account, seguendo le migliori pratiche prescrittive.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.

- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS SDK per Python (Boto3)è un kit di sviluppo software che ti aiuta a integrare la tua applicazione, libreria o script Python con. Servizi AWS
- AWS Service Catalogconsente di gestire centralmente i cataloghi dei servizi IT approvati. AWS Gli utenti finali possono distribuire rapidamente soltanto i servizi IT approvati di cui hanno bisogno, in accordo con i vincoli stabiliti dall'organizzazione.
- AWS Step Functionsè un servizio di orchestrazione senza server che consente di combinare AWS Lambda funzioni e altro Servizi AWS per creare applicazioni aziendali critiche.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS. Amazon VPC IP Address Manager (IPAM) è una funzionalità VPC che semplifica la pianificazione, il monitoraggio e il monitoraggio degli indirizzi IP per i carichi di lavoro. AWS

Altri strumenti

- GitHubè una piattaforma di sviluppo che gli sviluppatori possono utilizzare per creare, archiviare, gestire e condividere il proprio codice.
- HashiCorp Terraform è uno strumento Infrastructure as Code (IaC) che ti aiuta a creare e gestire risorse cloud e locali. Ciò include componenti di basso livello come istanze di calcolo, storage e rete e componenti di alto livello come voci DNS e funzionalità software a service (SaaS).
- Python è un linguaggio di programmazione per computer generico. È possibile utilizzarlo per creare applicazioni, automatizzare attività e sviluppare servizi su. Cloud AWS

Archivio di codici

- Il codice per questo pattern è disponibile nel repository GitHub AWS Control Tower Account Factory for Terraform.

Best practice

Quando si implementa AFT, si consiglia di seguire le migliori pratiche per garantire un'implementazione sicura, efficiente e di successo. Le linee guida e le raccomandazioni chiave per l'implementazione e il funzionamento di AFT includono quanto segue:

- Revisione approfondita degli input: esamina e comprendi attentamente ogni input. La corretta configurazione degli input è fondamentale per la configurazione e il funzionamento di AFT.
- Aggiornamenti regolari dei modelli: mantieni i modelli aggiornati con le ultime AWS funzionalità e le versioni di Terraform. Gli aggiornamenti regolari ti aiutano a sfruttare le nuove funzionalità e a mantenere la sicurezza.
- Controllo delle versioni: aggiungi la versione del modulo AFT e, se possibile, utilizza una distribuzione AFT separata per i test.
- Ambito: utilizzate AFT solo per implementare protezioni e personalizzazioni dell'infrastruttura. Non utilizzarlo per distribuire l'applicazione.
- Linting e convalida: la pipeline AFT richiede una configurazione Terraform linkata e convalidata. Esegui lint, convalida e testa prima di inviare la configurazione ai repository AFT.
- Moduli Terraform: crea codice Terraform riutilizzabile come moduli e specifica sempre le versioni di Terraform e del AWS provider in base ai requisiti della tua organizzazione.

Epiche

Configura e configura il tuo ambiente AWS

Operazione	Description	Competenze richieste
Implementazione. AWS Control Tower	Imposta e configura AWS Control Tower nel tuo AWS ambiente per garantire la gestione e la governance centralizzate del tuo. Account AWS Per ulteriori informazioni, consulta la sezione Guida introduttiva AWS Control Tower nella AWS Control Tower documentazione.	Amministratore cloud

Operazione	Description	Competenze richieste
Implementa AWS Control Tower Account Factory for Terraform (AFT).	Configura AFT in un nuovo account di gestione AFT dedicato. Per ulteriori informazioni, vedi Configurare e avviare AWS Control Tower Account Factory for Terraform nella AWS Control Tower documentazione.	Amministratore cloud
AFT completo dopo l'impleme ntazione.	Una volta completata la distribuzione dell'infrastruttur a AFT, completa i passaggi indicati nelle fasi successive alla distribuzione nella AWS Control Tower documenta zione.	Amministratore cloud

Crea IPAM

Operazione	Description	Competenze richieste
Delegare un amministratore IPAM.	Per delegare un account amministratore IPAM nella tua AWS organizzazione, utilizza i seguenti passaggi: 1. Utilizzando l'account di AWS Organizations gestione, apri la console IPAM all'indirizzo. https://console.aws.amazon.com/ipam/ 2. In Console di gestione AWS, scegli quello Regione	Amministratore AWS

Operazione	Description	Competenze richieste
	AWS in cui desideri lavorare con IPAM. 3. Nel riquadro di navigazione, seleziona Impostazioni organizzazione. 4. Scegli Delega. L'opzione Delega è disponibile solo se hai effettuato l'accesso alla console come account di AWS Organizations gestione. 5. Inserisci l' Account AWS ID di un account IPAM. L'amministratore IPAM deve avere un account AWS Organizations membro. 6. Scegli Save changes (Salva modifiche). In alternativa, è possibile utilizzare AWS CLI ed eseguire il seguente comando: `aws ec2 enable-ipam-organization-admin-account \ --delegated-admin-account-id 012345678901` Per ulteriori informazioni, consulta Integrare IPAM con gli account di un'	

Operazione	Description	Competenze richieste
	AWS organizzazione nella documentazione di Amazon VPC enable-ipam-organization-admine -account nel AWS CLI Command Reference. **Important** Per continuare a utilizzare IPAM, devi accedere all'account amministratore delegato. Il profilo SSO o le variabili di AWS ambiente specificate nel passaggio successivo devono consentire all'utente di accedere a tale account e concedere le autorizzazioni per creare un pool IPAM di primo livello e regionale.	

Operazione	Description	Competenze richieste

Crea un pool IPAM di primo livello e regionale.	Il GitHub repository di questo pattern contiene un modello Terraform che puoi utilizzare per creare il tuo pool IPAM di primo livello e il pool regionale . Quindi puoi condividere i pool con un'organizzazione, un'unità organizzativa (OU) o un'altra risorsa utilizzando (). Account AWS AWS Resource Access Manager AWS RAM
	
	Utilizza le fasi seguenti:
	
	1. Esegui i comandi seguenti:
	
	`# Navigate to the IPAM module`
	`cd ipam/terraform`
	
	`# Initiate the IPAM module`
	`terraform init`
	
	2. Aprite il `main.tf` file, modificate il codice come descritto nei commenti seguenti e salvate il file.
	
	`terraform { required_providers { aws = { source = "hashicorp/aws" version = ">= 5.11.0, < 6.0.0" }`
		Amministratore AWS

Operazione	Description	Competenze richieste

	`  }`
	`}`
	
	`provider "aws" {`
	`  region = "us-east-1" # Replace with the region to create your top-level pool`
	`}`
	
	`module "ipam" {`
	`  source  = "aws-ia/ipam/aws"`
	`  version = ">= 2.1.0"`
	
	`  top_name = "my-top-level-pool" # Replace with your desired top-level pool name`
	`  top_description = "top-level pool" # Replace with your desired top-level level pool description`
	`  top_cidr = ["10.0.0.0/16"] # Replace with your desired top-level pool CIDR`
	
	`  pool_configurations = {`
	`    my-regional-pool = { # (Optional) Replace with a different resource name`
	`      name = "my-regional-pool" # Replace with your desired pool name`	

Operazione	Description	Competenze richieste
	description = "regional pool" # Replace with your desired pool description cidr = ["10.0.0.0/23"] # Replace with your desired pool CIDR locale = "us-east-1" # Replace with your desired pool locale ram_share_principals = ["arn:aws:organizations::012345678901:ou/ou-ab1c2de345/ou-ab1c2de345"] # Replace with your desired principal ARN to share with via Resource Access Manager (RAM) } } } output "my_regional_pool_id" { description = "The ID of the regional pool" value = module.ipam.pools_level_1["my-regional-pool"].id # Replace with your desired resource name if changed above }	

Operazione	Description	Competenze richieste

	`# Create the IPAM top-level and resource pool`
	`terraform apply`
	
	Prendi nota dell'ID del pool di risorse che viene emesso dopo la creazione. L'ID ti servirà al momento dell'invio della richiesta di account. Se si dimentica l'ID del pool di risorse, è possibile recuperarlo in un secondo momento da. Console di gestione AWS
	
	Important
	Assicurati che i pool creati CIDRs non si sovrappongano a nessun altro pool nella tua regione di lavoro. Puoi creare un pool senza CIDR, ma non sarai in grado di utilizzare il pool per le assegnazioni fino a quando non avrai eseguito il provisioning di un CIDR. Puoi aggiungere qualcosa CIDRs a un pool in qualsiasi momento modificando il pool.	

Integra IPAM con AFT

Operazione	Description	Competenze richieste

Inizia a creare la personalizzazione dell'account.	Per iniziare una nuova personalizzazione dell'account, esegui i seguenti comandi dal tuo terminale:
	`# Default name for customization repo`
	`cd aft-account-customizations # Replace with your actual repo name if different than the default`
	`mkdir -p APG-AFT-IPAM/terraform # Replace APG-AFT-IPAM with your desired customization name`
	`cd APG-AFT-IPAM/terraform`
		DevOps ingegnere

Crea `aft-providers.jinja` file.	Aggiungi codice dinamico al `aft-providers.jinja` file che specifica il backend e il provider Terraform da utilizzare. Utilizza le fasi seguenti: 1. Crea un nuovo `aft-providers.jinja` file nella directory. `terraform` 2. Apri il file, aggiungi il codice seguente e salva il file. `provider "aws" {`	DevOps ingegnere

Operazione	Description	Competenze richieste

	<pre> region =
	 "{{ provider_
	region }}"
	  assume_role {
	  role_arn
	 = "{{ target_ad
	min_role_arn }}"
	  }
	  default_tags {
	  tags = {
	  managed_by
	  = "AFT"
	  }
	  }
	}</pre>	

Operazione	Description	Competenze richieste

Crea `backend.jinja` file.	Aggiungi codice dinamico al `backend.jinja` file che specifica il backend e il provider Terraform da utilizzare.
	
	Utilizza le fasi seguenti:
	
	1. Crea un nuovo `backend.jinja` file nella directory. `terraform`
	2. Apri il file, aggiungi il codice seguente e salva il file.
	
	`## Auto generated backend.tf ## ## Updated on: {{ timestamp }} ##`
	
	`{% if tf_distribution_type == "oss" -%} terraform { required_version = ">= {{ tf_version }}" backend "s3" { region = "{{ region }}" bucket = "{{ bucket }}" key = "{{ key }}" dynamodb_table = "{{ dynamodb_table }}" encrypt = "true"`
		DevOps ingegnere

Operazione	Description	Competenze richieste

	kms_key_id = "{{ kms_key_id }}"
	role_arn = "{{ aft_admin_role_arn }}"
	}
	}
	{% else -%}
	terraform {
	backend "remote" {
	organization = "{{ terraform_org_name }}"
	workspaces {
	name = "{{ terraform_workspace_name }}"
	}
	}
	}
	{% endif %}	

Operazione	Description	Competenze richieste

Crea `main.tf` file.	Crea un nuovo `main.tf` file e aggiungi codice che definisce due fonti di dati che recuperano due valori da AWS Systems Manager (`aws_ssm`) e crea il VPC.
	
	Utilizza le fasi seguenti:
	
	1. Crea un nuovo `main.tf` file nella directory. `terraform`
	2. Apri il file, aggiungi il codice seguente e salva il file.
	
	`# Define data sources`
	`data "aws_ssm_parameter" "vpc_ipam_id" {`
	`  name = "/aft/account-request/custom-fields/vpc-ipam-id" # Value is defined in the account-request.tf file`
	`}`
	
	`data "aws_ssm_parameter" "vpc_netmask" {`
	`  name = "/aft/account-request/custom-fields/vpc-ipam-netmask" # Value is defined in the account-request.tf file`
	`}`
		DevOps ingegnere

Operazione	Description	Competenze richieste

	`# Create new VPC resource "aws_vpc" "vpc1" {`
	
	`ipv4_ipam_pool_id = data.aws_ssm_parameter.vpc_ipam_id.value # Retrieved from SSM - this is how we integrate with IPAM`
	`ipv4_netmask_length = data.aws_ssm_parameter.vpc_netmask.value # Retrieved from SSM`
	
	`assign_generated_ipv6_cidr_block = var.enable_ipv6 ? true : null`
	`ipv6_cidr_block = var.ipv6_cidr`
	`ipv6_ipam_pool_id = var.ipv6_ipam_pool_id`
	`ipv6_netmask_length = var.ipv6_netmask_length`
	`ipv6_cidr_block_network_border_group = var.ipv6_cidr_block_network_border_group`
	
	`instance_tenancy = var.instance_tenancy`	

Operazione	Description	Competenze richieste

	<pre> enable_dns_hostnam
	es =
	 var.enable_dns_hos
	tnames
	  enable_dns_support
	                =
	 var.enable_dns_sup
	port
	  enable_network_add
	ress_usage_metrics =
	 var.enable_network
	_address_usage_met
	rics
	
	  tags = var.tags
	
	  lifecycle {
	    ignore_changes =
	 [
	     tags, # Any
	 changes made to VPC
	 tags after creation
	 will not be overwritt
	en - remove to revert
	 these changes during
	 future 'terraform
	 apply' operations
	    ]
	  }
	}</pre>	

Operazione	Description	Competenze richieste

Crea `variables.tf` file.	Crea un `variables.tf` file che dichiari le variabili utilizzate dal modulo Terraform.
	
	Utilizza le fasi seguenti:
	
	1. Crea un nuovo `variables.tf` file nella directory. `terraform`
	2. Apri il file, aggiungi il codice seguente e salva il file.
	
	`# Copied from AWS VPC module`
	`# https://github.com/terraform-aws-modules/terraform-aws-vpc/blob/master/variables.tf`
	
	`variable "name" {`
	`  description = "Name to be used on all the resources as identifier"`
	`  type        = string`
	`  default     = ""`
	`}`
	
	`variable "enable_ipv6" {`
	`  description = "Requests an Amazon-provided IPv6 CIDR block with a /56 prefix length for the VPC. You cannot specify the range of IP addresses, or`
		DevOps ingegnere

Operazione	Description	Competenze richieste

	the size of the CIDR block"
	type = bool
	default = false
	}
	
	variable "ipv6_cidr" {
	description = "(Optional) IPv6 CIDR block to request from an IPAM Pool. Can be set explicitly or derived from IPAM using `ipv6_netmask_length`"
	type = string
	default = null
	}
	
	variable "ipv6_ipam_pool_id" {
	description = "(Optional) IPAM Pool ID for a IPv6 pool. Conflicts with `assign_generated_ipv6_cidr_block`"
	type = string
	default = null
	}
	
	variable "ipv6_netmask_length" {
	description = "(Optional) Netmask length to request from IPAM Pool. Conflicts with `ipv6_cidr_block`. This can be omitted if IPAM pool as a	

Operazione	Description	Competenze richieste

	`allocation_default_netmask_length` set. Valid values: `56`"
	type = number
	default = null
	}
	
	variable "ipv6_cidr_block_network_border_group" {
	description = "By default when an IPv6 CIDR is assigned to a VPC a default ipv6_cidr_block_network_border_group will be set to the region of the VPC. This can be changed to restrict advertisement of public addresses to specific Network Border Groups such as LocalZones"
	type = string
	default = null
	}
	
	variable "instance_tenancy" {
	description = "A tenancy option for instances launched into the VPC"
	type = string
	default = "default"
	}
	
	variable "enable_dns_hostnames" {	

Operazione	Description	Competenze richieste

	<pre> description =
	 "Should be true to
	 enable DNS hostnames
	 in the VPC"
	  type = bool
	  default = true
	}
	
	variable "enable_d
	ns_support" {
	  description =
	 "Should be true to
	 enable DNS support in
	 the VPC"
	  type = bool
	  default = true
	}
	
	variable "enable_n
	etwork_address_usa
	ge_metrics" {
	  description =
	 "Determines whether
	 network address usage
	 metrics are enabled
	 for the VPC"
	  type = bool
	  default = null
	}
	
	variable "tags" {
	  description = "A
	 map of tags to add to
	 all resources"
	  type =
	 map(string)
	  default = {}
	}</pre>	

Operazione	Description	Competenze richieste

Crea `terraform.tfvars` file.	Crea un `terraform .tfvars` file che definisca i valori delle variabili che vengono passate al `main.tf` file.
	
	Utilizza le fasi seguenti:
	
	1. Crea un nuovo `terraform .tfvars` file nella `terraform` directory.
	2. Apri il file, aggiungi il codice seguente e salva il file.
	
	`name = "demo-ipam"`
	`enable_ipv6 = false`
	`enable_dns_hostnames = true`
	`enable_dns_support = true`
		DevOps ingegnere

Operazione	Description	Competenze richieste

Crea `outputs.tf` file.	Crea un nuovo `outputs.tf` file che esponga alcuni valori in CodeBuild.
	
	Utilizza le fasi seguenti:
	
	1. Crea un nuovo `outputs.tf` file nella `terraform` directory.
	2. Apri il file, aggiungi il codice seguente e salva il file.
	
	`# Output VPC ID and ARN in CodeBuild`
	`output "vpc1" {`
	`  description = "VPC 1 information"`
	`  value = {`
	`  id  = try(aws_vpc.vpc1.id, null)`
	`  arn = try(aws_vpc.vpc1.arn, null)`
	`  }`
	`}`
		DevOps ingegnere

Operazione	Description	Competenze richieste

Effettua la personalizzazione.	Per salvare la nuova personalizzazione nell'archivio delle personalizzazioni dell'account, esegui i seguenti comandi:
	`# Assumes you are still in the /terraform directory`
	`cd .. # Skip if you are in the account customization root directory (APG-AFT-IPAM)`
	`git add .`
	`git commit -m "APG customization"`
	`git push origin`
		DevOps ingegnere

Operazione	Description	Competenze richieste
Applica la personalizzazione.	Aggiungi il codice al `account-requests.tf` file che richiede un nuovo account con la personalizzazione dell'account appena creato. I campi personalizzati creano i parametri di Systems Manager nell'account fornito necessari per creare il VPC con il CIDR allocato IPv4 IPAM corretto. Utilizza le fasi seguenti: 1. Dal tuo terminale, accedi alla directory. `aft-account-request/terraform` 2. Apri il `account-requests.tf` file, aggiungi e modifica il codice seguente con i tuoi valori e salva il file. `module "sandbox_account_01" { source = "./modules/aft-account-request" control_tower_parameters = { AccountEmail = "john.doe@example.com"`	AWS DevOps

Operazione	Description	Competenze richieste

	AccountName = "sandbox-account-01"
	# Syntax for top-level OU
	ManagedOrganizationalUnit = "Sandbox"
	# Syntax for nested OU
	# ManagedOrganizationalUnit = "Sandbox (ou-xfe5-a8hb8ml8)"
	SSOUserEmail = "john.doe@example.com"
	SSOUserFirstName = "John"
	SSOUserLastName = "Doe"
	}
	
	account_tags = {
	"ABC:Owner" = "john.doe@example.com"
	"ABC:Division" = "ENT"
	"ABC:Environment" = "Dev"
	"ABC:CostCenter" = "123456"
	"ABC:Vended" = "true"
	"ABC:DivCode" = "102"
	"ABC:BUCode" = "ABC003"
	"ABC:Project" = "123456"
	}	

Operazione	Description	Competenze richieste

```
  change_management_
parameters = {
    change_re
quested_by = "John
 Doe"
    change_reason
       = "testing
 the account vending
 process"
  }

  custom_fields = {
    vpc-ipam-id =
 "ipam-pool-123456a
b789041cd2"
    vpc-ipam-netmask
 = "24"

  }

  account_customizat
ions_name = "APG-AFT-
IPAM" # Do not change
 this value
}
```

> **Important**
>
> Assicurati che il `vpc-ipam-netmask` valore sia inferiore o uguale all'allocazione `netmask` del pool IPAM e che nel pool IPAM sia IPs disponibile una quantità sufficiente

Operazione	Description	Competenze richieste

	per l'allocazione al VPC.
	3. Per salvare il nuovo account nell'archivio delle richieste di account, utilizza i seguenti comandi:
	`git add .`
	`git commit -m 'requested APG account'`
	`git push origin`	

Operazione	Description	Competenze richieste
Convalida la personalizzazione.	Accedi all'account appena venduto e verifica che la personalizzazione sia stata applicata correttamente. Utilizza le fasi seguenti: 1. Apri la pagina di AWS IAM Identity Center accesso ed esegui l'autenticazione come utente specificato nel file. `account-requests.tf` 2. Dall'elenco degli account IAM Identity Center, scegli l'account che hai venduto in precedenza e accedi utilizzando il ruolo. AWS `AdministratorAccess` 3. Vai alla console VPC e seleziona Your. VPCs 4. Seleziona l'ID VPC del VPC che hai appena creato. 5. Scegli la scheda CIDRs. Dovresti vedere un CIDR allocato tramite IPAM IPv4 . 6. (Facoltativo) Accedi al tuo account amministratore delegato IPAM e verifica che l'allocazione sia mostrata nella console IPAM.	DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
Si verificano errori nella creazione o nella gestione delle risorse causati da autorizzazioni inadeguate.	Esamina i ruoli e le policy AWS Identity and Access Management (IAM) associati a Step Functions e agli altri servizi coinvolti nella distribuzione. CodeBuild Verifica che dispongan o delle autorizzazioni necessarie. In caso di problemi di autorizzazione, modifica le politiche IAM per concedere l'accesso richiesto.
Le Servizio AWS quote vengono raggiunte durante la distribuzione.	Prima di distribuire la pipeline, controlla le Servizio AWS quote per risorse come i bucket Amazon Simple Storage Service (Amazon S3), i ruoli e le funzioni IAM. AWS Lambda Se necessario, richiedi un aumento delle quote. Per ulteriori informazioni, consulta Servizio AWS Quotas nella documentazione generale di riferimento di AWS .

Risorse correlate

Servizio AWS documentazione

- AWS Control Tower Guida per l'utente
- Come funziona IPAM
- Best practice per la sicurezza in IAM
- Servizio AWS quote

Altre risorse

- documentazione Terraform Provider AWS

Automatizza l'aggiunta o l'aggiornamento delle voci di registro di Windows utilizzando AWS Systems Manager

Appasaheb Bagali, Amazon Web Services

Riepilogo

AWS Systems Manager è uno strumento di gestione remota per le istanze Amazon Elastic Compute Cloud (Amazon EC2). Systems Manager offre visibilità e controllo sulla tua infrastruttura su Amazon Web Services. Questo strumento versatile può essere utilizzato per correggere le modifiche del registro di Windows identificate come vulnerabilità dal rapporto di scansione delle vulnerabilità di sicurezza.

Questo schema illustra i passaggi per proteggere le EC2 istanze che eseguono il sistema operativo Windows automatizzando le modifiche al registro consigliate per la sicurezza dell'ambiente. Il pattern utilizza il comando Run per eseguire un documento Command. Il codice è allegato e una parte di esso è inclusa nella sezione Codice.

Prerequisiti e limitazioni

- Un account AWS attivo
- Autorizzazioni per accedere all' EC2 istanza e Systems Manager

Architecture

Stack tecnologico Target

- Un cloud privato virtuale (VPC), con due sottoreti e un gateway NAT (Network Address Translation)
- Un documento Systems Manager Command per aggiungere o aggiornare il nome e il valore del registro
- Systems Manager Run Command per eseguire il documento Command sulle EC2 istanze specificate

Architettura Target

Tools (Strumenti)

Strumenti

- Politiche e ruoli IAM: AWS Identity and Access Management (IAM) è un servizio Web che ti aiuta a controllare in modo sicuro l'accesso alle risorse AWS. Utilizza IAM per controllare chi è autenticato (accesso effettuato) e autorizzato (dispone di autorizzazioni) per l'utilizzo di risorse.
- Amazon Simple Storage Service — Amazon Simple Storage Service (Amazon S3) è uno storage per Internet. È concepito per rendere più accessibili agli sviluppatori risorse informatiche su grande scala per il Web. In questo modello, viene utilizzato un bucket S3 per archiviare i log di Systems Manager.
- AWS Systems Manager — AWS Systems Manager è un servizio AWS che puoi usare per visualizzare e controllare la tua infrastruttura su AWS. Systems Manager ti aiuta a mantenere la sicurezza e la conformità scansionando le istanze gestite e segnalando (o adottando misure correttive) eventuali violazioni delle policy rilevate.
- Documento AWS Systems Manager Command: i documenti AWS Systems Manager Command vengono utilizzati da Run Command. La maggior parte dei documenti Command è supportata su tutti i sistemi operativi Linux e Windows Server supportati da Systems Manager.
- AWS Systems Manager Run Command — AWS Systems Manager Run Command ti offre un modo per gestire la configurazione delle istanze gestite in remoto e in sicurezza. Utilizzando Run Command, puoi automatizzare le attività amministrative più comuni ed eseguire modifiche di configurazione una tantum su larga scala.

Codice

È possibile utilizzare il codice di esempio seguente per aggiungere o aggiornare un nome di registro di Microsoft Windows in`Version`, un percorso del Registro di `HKCU:\Software\ScriptingGuys\Scripts` sistema e un valore in`2`.

```
#Windows registry path which needs to add/update
$registryPath ='HKCU:\\Software\\ScriptingGuys\\Scripts'
#Windows registry Name  which needs to add/update
$Name = 'Version'
#Windows registry value  which needs to add/update
$value = 2
# Test-Path cmdlet to see if the registry key exists.
IF(!(Test-Path $registryPath))
        {
```

```
          New-Item -Path $registryPath -Force | Out-Null
          New-ItemProperty -Path $registryPath -Name $name -Value     $value ` -
PropertyType DWORD -                 Force | Out-        Null
        } ELSE {
                      New-ItemProperty -Path $registryPath -Name $name -Value $value `
 -PropertyType            DWORD        -Force | Out-Null
           }
echo 'Registry Path:'$registryPath
 echo 'Registry Name:'$registryPath
 echo 'Registry Value:'(Get-ItemProperty -Path $registryPath -Name $Name).version
```

È allegato l'esempio di codice JSON (JavaScript Object Notation) completo del documento Systems Manager Command.

Epiche

Configurazione di un VPC

Operazione	Description	Competenze richieste
Crea un VPC.	Nella Console di gestione AWS, crea un VPC con sottoreti pubbliche e private e un gateway NAT. Per ulteriori informazioni, consulta la documentazione di AWS.	Amministratore cloud
Crea gruppi di sicurezza.	Assicurati che ogni gruppo di sicurezza consenta l'accesso a Remote Desktop Protocol (RDP) dall'indirizzo IP di origine.	Amministratore cloud

Crea una policy IAM e un ruolo IAM

Operazione	Description	Competenze richieste
Creare una policy IAM	Crea una policy IAM che fornisca l'accesso ad Amazon	Amministratore cloud

Operazione	Description	Competenze richieste
	S3 EC2, Amazon e Systems Manager.	
Crea un ruolo IAM.	Crea un ruolo IAM e collega la policy IAM che fornisce l'accesso ad Amazon S3 EC2, Amazon e Systems Manager.	Amministratore cloud

Esegui l'automazione

Operazione	Description	Competenze richieste
Creare il documento Systems Manager Command.	Crea un documento Systems Manager Command che distribuirà le modifiche del registro di Microsoft Windows da aggiungere o aggiornare.	Amministratore cloud
Esegui il comando Systems Manager Run.	Eseguite il comando Systems Manager Run, seleziona ndo il documento Command e le istanze di destinazi one di Systems Manager. Ciò invia la modifica del registro di Microsoft Windows nel documento Command selezionato alle istanze di destinazione.	Amministratore cloud

Risorse correlate

- AWS Systems Manager
- Documenti AWS Systems Manager
- Comando di esecuzione di AWS Systems Manager

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Crea automaticamente un RFC in AMS usando Python

Gnanasekaran Kailasam, Amazon Web Services

Riepilogo

AWS Managed Services (AMS) ti aiuta a gestire la tua infrastruttura basata sul cloud in modo più efficiente e sicuro fornendo una gestione continua dell'infrastruttura Amazon Web Services (AWS). Per apportare una modifica al tuo ambiente gestito, devi creare e inviare una nuova richiesta di modifica (RFC) che includa un ID del tipo di modifica (CT) per una particolare operazione o azione.

Tuttavia, la creazione manuale di una RFC può richiedere circa cinque minuti e i team dell'organizzazione potrebbero doverne inviare più di una RFCs ogni giorno. Questo modello consente di automatizzare il processo di creazione RFC, ridurre i tempi di creazione per ogni RFC ed eliminare gli errori manuali.

Questo modello descrive come utilizzare il codice Python per creare automaticamente la `Stop EC2 instance` RFC che blocca le istanze Amazon Elastic Compute Cloud (Amazon EC2) nel tuo account AMS. È quindi possibile applicare l'approccio di questo modello e l'automazione Python ad altri tipi di RFC.

Prerequisiti e limitazioni

Prerequisiti

- Un account AMS Advanced. Per ulteriori informazioni a riguardo, consulta i piani operativi di AMS nella documentazione di AWS Managed Services.
- Almeno un'istanza EC2 esistente nel tuo account AMS.
- Una comprensione di come creare e inviare RFCs in AMS.
- Familiarità con Python.

Limitazioni

- Puoi utilizzarlo solo RFCs per le modifiche nel tuo account AMS. Il tuo account AWS utilizza processi diversi per modifiche simili.

Architecture

Stack tecnologico

- ARMS
- Interfaccia a riga di comando di AWS (CLI AWS)
- AWS SDK per Python (Boto3)
- Python e i suoi pacchetti richiesti (JSON e Boto3)

Automazione e scalabilità

Questo pattern fornisce un codice di esempio per automatizzare la `Stop EC2 instance` RFC, ma è possibile utilizzare il codice di esempio e l'approccio di questo pattern per altri scopi. RFCs

Tools (Strumenti)

- AWS Managed Services — AMS ti aiuta a gestire la tua infrastruttura AWS in modo più efficiente e sicuro.
- AWS CLI — AWS Command Line Interface (AWS CLI) è uno strumento unificato per gestire i servizi AWS. In AMS, l'API di gestione delle modifiche fornisce operazioni di creazione e gestione. RFCs
- SDK AWS per Python (Boto3) — SDK per Python semplifica l'integrazione di applicazioni, librerie o script Python con i servizi AWS.

Codice

Il `AMS Stop EC2 Instance.zip` file (allegato) contiene il codice Python per creare un `Stop EC2 instance` RFC. Puoi anche configurare questo codice per inviare una singola RFC per più istanze EC2.

Epiche

Opzione 1: configurare l'ambiente per macOS o Linux

Operazione	Description	Competenze richieste
Installa e convalida Python.	1. Apri una finestra di terminale ed esegui il `brew install python3` comando.	Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	2. Verifica che Python sia installato correttamente `python --version` eseguendo il comando. 3. Verifica che `pip` sia installato correttamente eseguendo il comando. `pip --version`	
Installa AWS CLI.	Esegui il *`pip install awscli --upgrade –user`* comando per installare AWS CLI.	Amministratore di sistema AWS
Installa Boto3.	Esegui il `pip install boto3` comando per installare Boto3.	Amministratore di sistema AWS
Installa JSON.	Esegui il `pip install json` comando per installare JSON.	Amministratore di sistema AWS
Configura AMS CLI.	Accedi alla Console di gestione AWS, apri la console AMS e scegli Documenta tion. Scaricate il file.zip che contiene la CLI AMS, decomprimetelo e installatelo sul computer locale. Dopo aver installato AMS CLI, esegui il `aws amscm help` comando. L'output fornisce informazioni sul processo di gestione delle modifiche di AMS.	Amministratore di sistema AWS

Opzione 2: configurazione dell'ambiente per Windows

Operazione	Description	Competenze richieste
Installa e convalida Python.	1. Apri la pagina delle versioni di Python per Windows, scarica la versione più recente e installa Python. 2. Verifica che Python sia installato correttamente `python --version` eseguendo il comando. 3. Verifica che `pip` sia installato correttamente eseguendo il comando. `pip --version`	Amministratore di sistema AWS
Installa AWS CLI.	Esegui il `pip install awscli --upgrade –user` comando per installare AWS CLI.	Amministratore di sistema AWS
Installa Boto3.	Esegui il `pip install boto3` comando per installare Boto3.	Amministratore di sistema AWS
Installa JSON.	Esegui il `pip install json` comando per installare JSON.	Amministratore di sistema AWS
Configura AMS CLI.	Accedi alla Console di gestione AWS, apri la console AMS e scegli Documenta tion. Scaricate il file.zip che contiene la CLI AMS, decomprimetelo e installatelo sul computer locale.	Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	Dopo aver installato AMS CLI, esegui il `aws amscm help` comando. L'output fornisce informazioni sul processo di gestione delle modifiche di AMS	

Estrai l'ID CT e i parametri di esecuzione per l'RFC

Operazione	Description	Competenze richieste
Estrai l'ID CT, la versione e i parametri di esecuzione per l'RFC.	Ogni RFC ha un ID CT, una versione e parametri di esecuzione diversi. È possibile estrarre queste informazioni utilizzando una delle seguenti opzioni: 1. Segui le istruzioni della sezione Finding a request for change (RFC) with the CLI negli esempi di utilizzo di RFC tratti dalla documentazione di AWS Managed Services. 2. Apri un RFC esistente di tipo simile o crea un nuovo RFC come test tramite la console AMS. Utilizza l'ID CT e i parametri di esecuzione di RFC. Per ulteriori informazioni a riguardo, consulta Finding an RFC with the console	Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	nella documentazione di AWS Managed Services. **Note** Per adattare l'automazione Python di questo pattern ad altre RFCs, sostituisci il tipo CT e i valori dei parametri nel file di codice `ams_stop_ec2_instance` Python dal `AMS Stop EC2 Instance.zip` file (allegato) con quelli che hai estratto.	

Esegui l'automazione Python

Operazione	Description	Competenze richieste
Esegui l'automazione Python.	1. Scarica il `AMS Stop EC2 Instance.zip` file (allegato) sul tuo computer locale ed estrai il file. 2. Aggiorna `input_instances` con le informazioni sulla tua istanza EC2. 3. Apri un terminale e vai al percorso del codice estratto	Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	4. Esegui il comando `pythonams_stop_ec2_instance.py .`	

Risorse correlate

- Cosa sono i tipi di modifica?
- Tutorial CLI: stack a due livelli ad alta disponibilità (Linux/RHEL)

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Arresta e avvia automaticamente un'istanza database Amazon RDS utilizzando AWS Systems Manager Maintenance Windows

Ashita Dsilva, Amazon Web Services

Riepilogo

Questo modello dimostra come arrestare e avviare automaticamente un'istanza DB di Amazon Relational Database Service (Amazon RDS) secondo una pianificazione specifica (ad esempio, chiudere un'istanza DB al di fuori dell'orario di lavoro per ridurre i costi) utilizzando Maintenance Windows. AWS Systems Manager A tal fine, Systems Manager è conveniente per i casi d'uso tipici.

AWS Systems Manager L'automazione fornisce i `AWS-StopRdsInstance` `AWS-StartRdsInstance` runbook per arrestare e avviare le istanze database di Amazon RDS. Ciò significa che non è necessario scrivere una logica personalizzata con AWS Lambda funzioni o creare una regola Amazon CloudWatch Events.

Systems Manager offre due funzionalità per la pianificazione delle attività: State Manager e Maintenance Windows. State Manager imposta e mantiene la configurazione dello stato richiesta per le risorse nel tuo account Amazon Web Services (AWS) una sola volta o secondo una pianificazione specifica. Manutenzione Windows esegue le attività sulle risorse del tuo account durante una finestra temporale specifica. Sebbene sia possibile utilizzare l'approccio di questo modello con State Manager o Maintenance Windows, si consiglia di utilizzare Maintenance Windows perché può eseguire una o più attività in base alla priorità assegnata e può anche eseguire AWS Lambda funzioni e AWS Step Functions attività. Per ulteriori informazioni su State Manager e Maintenance Windows, vedere Scelta tra State Manager e Maintenance Windows nella documentazione di Systems Manager.

Questo modello fornisce passaggi dettagliati per configurare due finestre di manutenzione separate che utilizzano espressioni cron per interrompere e quindi avviare un'istanza database Amazon RDS.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- Un'istanza database Amazon RDS esistente che desideri interrompere e avviare secondo una pianificazione specifica.
- Espressioni Cron per la pianificazione richiesta. Ad esempio, l'espressione `cron(0 9 ? * MON-FRI *)` esegue l'attività alle 09:00 ogni lunedì, martedì, mercoledì, giovedì e venerdì. Per

ulteriori informazioni, consulta Cron and rate expressions for maintenance nella documentazione di Systems Manager.

- Familiarità con Systems Manager.
- Autorizzazioni per avviare e arrestare l'istanza RDS. Per ulteriori informazioni, consulta la sezione Epics.

Limitazioni

- Un'istanza database Amazon RDS può essere interrotta per un massimo di sette giorni alla volta. Dopo sette giorni, l'istanza DB si riavvia automaticamente per garantire che riceva gli aggiornamenti di manutenzione necessari.
- Non è possibile interrompere un'istanza DB che è una replica di lettura o che ha una replica di lettura.
- Non è possibile interrompere un'istanza DB di Amazon RDS for SQL Server in una configurazione Multi-AZ.
- Le quote di servizio si applicano a Maintenance Windows e Systems Manager Automation. Per ulteriori informazioni sulle quote di servizio, consulta AWS Systems Manager endpoint e quote nella documentazione. Riferimenti generali di AWS
- Alcune Servizi AWS non sono disponibili in tutte. Regioni AWS Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Architecture

Il diagramma seguente mostra il flusso di lavoro per arrestare e avviare automaticamente un'istanza database Amazon RDS.

Il flusso di lavoro prevede i seguenti passaggi:

1. Crea una finestra di manutenzione e usa le espressioni cron per definire la pianificazione di arresto e avvio per le tue istanze database Amazon RDS.

2. Registrare un'attività di Systems Manager Automation nella finestra di manutenzione utilizzando il `AWS-StartRdsInstance` runbook `AWS-StopRdsInstance` or.

3. Registra una destinazione nella finestra di manutenzione utilizzando un gruppo di risorse basato su tag per le tue istanze database Amazon RDS.

Stack tecnologico

- AWS CloudFormation
- AWS Identity and Access Management (IAM)
- Amazon RDS
- Systems Manager

Automazione e scalabilità

Puoi interrompere e avviare più istanze DB Amazon RDS contemporaneamente etichettando le istanze DB Amazon RDS richieste, creando un gruppo di risorse che include tutte le istanze DB con tag e registrando questo gruppo di risorse come destinazione per la finestra di manutenzione.

Tools (Strumenti)

- AWS CloudFormationè un servizio che ti aiuta a modellare e configurare le tue risorse. AWS
- AWS Identity and Access Management (IAM) è un servizio web che consente di controllare in modo sicuro l'accesso alle AWS risorse.
- Amazon Relational Database Service (Amazon RDS) è un servizio Web che semplifica la configurazione, il funzionamento e la scalabilità di un database relazionale in. Cloud AWS
- AWS Resource Groupsti aiuta a organizzare AWS le risorse in gruppi, etichettare le risorse e gestire, monitorare e automatizzare le attività su risorse raggruppate.
- AWS Systems Managerè uno Servizio AWS strumento che puoi usare per visualizzare e controllare la tua infrastruttura. AWS Questo modello utilizza le seguenti funzionalità di Systems Manager:
  - AWS Systems Manager L'automazione semplifica le attività comuni di manutenzione e distribuzione delle istanze Amazon Elastic Compute Cloud EC2 (Amazon) e di altre risorse. AWS
  - AWS Systems Manager Manutenzione Windows ti aiuta a definire una pianificazione per quando eseguire azioni potenzialmente dannose sulle tue istanze.

Epiche

Creare e configurare il ruolo del servizio IAM per Systems Manager Automation

Operazione	Description	Competenze richieste
Configura il ruolo del servizio IAM per Systems Manager Automation.	Accedere a Console di gestione AWS e creare un ruolo di servizio per Systems Manager Automation. È possibile utilizzare uno dei due metodi seguenti per creare questo ruolo di servizio: • Utilizzare AWS CloudFormation per configurare un ruolo di servizio per Systems Manager Automation • Usa IAM per configurare i ruoli per Systems Manager Automation Il flusso di lavoro Systems Manager Automation richiama Amazon RDS utilizzando un ruolo di servizio per eseguire azioni di avvio e arresto sull'istanza database di Amazon RDS. Il ruolo di servizio deve essere configurato con la seguente policy in linea che dispone delle autorizzazioni per avviare e arrestare l'istanza DB di Amazon RDS:	Amministratore AWS

Operazione	Description	Competenze richieste

	<pre>{
	    "Version":
	"2012-10-17",
	    "Statement": [
	        {
	            "Sid":
	"RdsStartStop",
	            "Effect":
	"Allow",
	            "Action": [
	
	"rds:StopDBInstance",
	
	"rds:StartDBInstance"
	            ],
	            "Resource":
	"<RDS_Instance_ARN>"
	
	        },
	        {
	            "Sid":
	"RdsDescribe",
	            "Effect":
	"Allow",
	            "Action":
	"rds:DescribeDBIns
	tances",
	            "Resource":
	"*"
	        }
	    ]
	}</pre> Assicurati di sostituirla `<RDS_Instance_ARN>` con l'Amazon Resource Name (ARN) della tua istanza database Amazon RDS.	

Operazione	Description	Competenze richieste
	Se non hai dimestichezza con l'uso delle policy e dei ruoli IAM, segui le istruzioni nella sezione Panoramica della soluzione del post di AWS Systems Manager blog Schedule Amazon RDS stop and start use. **Important** Assicurati di registrare l'ARN del ruolo di servizio.	

Crea un gruppo di risorse

Operazione	Description	Competenze richieste
Etichetta le istanze database di Amazon RDS.	Apri la console Amazon RDS e tagga le istanze database di Amazon RDS che desideri aggiungere al gruppo di risorse. Un tag è un metadato assegnato a una AWS risorsa e consiste in una coppia chiave-valore. Ti consigliamo di utilizzare Action come chiave Tag e StartStopcome Valore. Per ulteriori informazioni su questo argomento, consulta Aggiungere,	Amministratore AWS

Operazione	Description	Competenze richieste
	elencare e rimuovere tag nella documentazione di Amazon RDS.	
Crea un gruppo di risorse per le tue istanze database Amazon RDS con tag.	Apri la AWS Resource Groups console e crea un gruppo di risorse basato sul tag che hai creato per le tue istanze database Amazon RDS. In Grouping Criteria, assicurati di scegliere AWS: :RDS:: DBInstance per il tipo di risorsa, quindi fornisci la coppia chiave-valore del tag (ad esempio, «Action-»). StartStop Ciò garantisce che il servizio controlli solo le istanze database di Amazon RDS e non altre risorse con questo tag. Assicurati di registrare il nome del gruppo di risorse. Per ulteriori informazioni e passaggi dettagliati, consulta Creare una query basata su tag e creare un gruppo nella AWS Resource Groups documentazione.	Amministratore AWS

Configura una finestra di manutenzione per arrestare le istanze database di Amazon RDS

Operazione	Description	Competenze richieste
Creare una finestra di manutenzione.	1. Apri la console Systems Manager, scegli Maintenance Windows, quindi scegli Crea una finestra di manutenzione. Fornisci un nome per la finestra di manutenzione (ad esempio, "StopRdsInstance«), inserisci una descrizione, quindi deseleziona Consenti destinazioni non registrate. 2. Scegli l'espressione CRON/rate e fornisci l'espressione di pianificazione per definire quando le istanze DB di Amazon RDS devono essere interrotte. Inserisci 1 per la Durata e 0 per Interrompi l'avvio delle attività. Per impostazione predefinita, il fuso orario mostra l'UTC. È possibile modificare il fuso orario per avviare la finestra di manutenzione in base al timestamp definito nell'espressione cron. 3. Scegli Crea finestra di manutenzione. Il sistema riporta alla pagina della finestra di manutenzione	Amministratore AWS

Operazione	Description	Competenze richieste
	e lo stato della finestra di manutenzione è Abilitato. **Important** L'operazione di arresto dell'istanza DB viene eseguita quasi istantaneamente una volta avviata e non copre l'intera durata della finestra di manutenzione. Questo modello fornisce i valori minimi per Duration e Stop all'avvio delle attività, poiché sono i parametri richiesti per una finestra di manutenzione. Per ulteriori informazioni e passaggi dettagliati, vedere Creare una finestra di manutenzione (console) nella documentazione di Systems Manager.	

Operazione	Description	Competenze richieste
Assegna un obiettivo alla finestra di manutenzione.	1. Nella console Systems Manager, scegli Finestre di manutenzione, scegli Azioni, quindi scegli Registra obiettivi. 2. Nell'area Obiettivi, specifica Scegli un gruppo di risorse, quindi scegli il nome di un gruppo di risorse esistente nel tuo account. 3. Per i tipi di risorse, scegli AWS: :RDS:: DBInstance e poi scegli Register target. Per ulteriori informazioni e passaggi dettagliati, vedere Assegnare obiettivi a una finestra di manutenzione (console) nella documenta zione di Systems Manager.	Amministratore AWS

Operazione	Description	Competenze richieste
Assegna un'attività alla finestra di manutenzione.	1. Nella console Systems Manager, scegli Maintenance Windows, quindi scegli la finestra di manutenzione. Scegli Azioni, quindi scegli Registra attività di automazione. 2. Per Document, scegli AWS-StopRdsInstance. 3. Nella sezione Target, scegli Selezione dei gruppi target registrati, quindi scegli l'oggetto della finestra di manutenzione che hai registrato nella finestra di manutenzione corrente. 4. Per il controllo della frequenza, specificate il 100% per la soglia di concorrenza ed errore. È possibile modificare i valori di controllo della frequenza in base ai requisiti di concorrenza delle attività e alla soglia di errore. Per ulteriori informazioni su questo argomento, vedere Informazioni sulla concorrenza e sulle soglie di errore nella documentazione di Systems Manager. 5. Nella sezione Ruolo di servizio IAM, per Ruolo di servizio, lascia vuota questa	Amministratore AWS

Operazione	Description	Competenze richieste
	casella o crea il tuo ruolo personalizzato. Se si lascia la casella vuota, Systems Manager crea automatic amente il ruolo collegato al servizio AWSServic eRoleForAmazonSSM e quindi associa il ruolo all'attività. Per creare un ruolo personalizzato, consulta Creare un ruolo di servizio personalizzato per le finestre di manutenzione (console), quindi associare quel ruolo personalizzato all'attività. 6. Nella sezione Parametri di input, specifica i seguenti parametri per il runbook: • InstanceId: `{{RESOURC E_ID}}` **Note** Infatti InstanceI d, viene utilizzat o uno pseudo parametro per estrarre l'ID di risorsa Amazon RDS DB dall'ARN. Per ulteriori informazi oni sugli pseudo parametri, vedere	

Operazione	Description	Competenze richieste
	Informazioni sugli pseudo parametri nella documentazione di Systems Manager. • AutomationAssumeRole: Fornisci l'ARN del ruolo di servizio creato per Systems Manager Automation. 7. Scegli l'attività Registra automazione. **Important** L'opzione Ruolo di servizio definisce il ruolo di servizio richiesto per la finestra di manutenzione per eseguire le attività. Tuttavia, questo ruolo non è identico al ruolo di servizio creato in precedenza per Systems Manager Automation. Per ulteriori informazioni e passaggi dettagliati, vedere Assegnare attività a una finestra di manutenzione	

Operazione	Description	Competenze richieste
	(console) nella documenta zione di Systems Manager.	

Configura una finestra di manutenzione per avviare le istanze database di Amazon RDS

Operazione	Description	Competenze richieste
Configura una finestra di manutenzione per avviare le istanze database di Amazon RDS.	Ripeti i passaggi dalla finestra Configura una manutenzione per interrompere l'epic delle istanze Amazon RDS DB per configurare un'altra finestra di manutenzione per avviare le istanze Amazon RDS DB a un orario pianificato. **Important** È necessario apportare le seguenti modifiche quando si configura la finestra di manutenzione per avviare le istanze DB: • Usa un nuovo nome per la finestra di manutenzione (ad esempio, "StartRds Instance«). • Sostituisci l'espressione cron con l'espressione cron che desideri utilizzare per avviare le istanze DB.	Amministratore AWS

Operazione	Description	Competenze richieste
	• Sostituisci il **`AWS-StopRdsInstance`** runbook con in Task. **`AWS-StartRdsInstance`**	

Risorse correlate

- Usa i documenti di Systems Manager Automation per gestire le istanze e ridurre i costi fuori orario (AWS post sul blog)

Centralizza la distribuzione dei pacchetti software in AWS Organizations utilizzando Terraform

Pradip Kumar Pandey, Chintamani Aphale, TV.R.L.Phani Kumar Dadi, Pratap Kumar Nanda, Aarti Rajput e Mayuri Shinde, Amazon Web Services

Riepilogo

Le aziende spesso gestiscono Account AWS più file distribuiti su più file Regioni AWS per creare una forte barriera di isolamento tra i carichi di lavoro. Per garantire la sicurezza e la conformità, i team di amministrazione installano strumenti basati su agenti come CrowdStrike, o TrendMicrostrumenti per la scansione di sicurezza SentinelOne, e l' CloudWatch agente Amazon, l'agenteDatadog o gli agenti per il monitoraggio. AppDynamics Questi team spesso incontrano difficoltà quando vogliono automatizzare centralmente la gestione e la distribuzione dei pacchetti software in questo ampio panorama.

Distributor, una funzionalità di AWS Systems Manager, automatizza il processo di creazione di pacchetti e pubblicazione del software per le istanze Microsoft Windows e Linux gestite sul cloud e sui server locali tramite un'unica interfaccia semplificata. Questo modello dimostra come è possibile utilizzare Terraform per semplificare ulteriormente il processo di gestione dell'installazione del software e per eseguire script su un gran numero di istanze e account membri all'interno con il minimo sforzo. AWS Organizations

Questa soluzione funziona per le istanze Amazon, Linux e Windows gestite da Systems Manager.

Prerequisiti e limitazioni

- Un pacchetto Distributor contenente il software da installare
- Terraform versione 0.15.0 o successiva
- Istanze Amazon Elastic Compute Cloud (Amazon EC2) gestite da Systems Manager e dotate di autorizzazioni di base per accedere ad Amazon Simple Storage Service (Amazon S3) nell'account di destinazione
- Una landing zone per la tua organizzazione configurata utilizzando AWS Control Tower
- (Opzionale) Account Factory for Terraform (AFT)

Architecture

Dettagli delle risorse

Questo modello utilizza Account Factory for Terraform (AFT) per creare tutte le AWS risorse richieste e la pipeline di codice per distribuire le risorse in un account di distribuzione. La pipeline di codice viene eseguita in due repository:

- La personalizzazione globale contiene il codice Terraform che verrà eseguito su tutti gli account registrati con AFT.
- Le personalizzazioni dell'account contengono il codice Terraform che verrà eseguito nell'account di distribuzione.

È inoltre possibile distribuire questa soluzione senza utilizzare AFT, eseguendo i comandi Terraform nella cartella delle personalizzazioni dell'account.

Il codice Terraform distribuisce le seguenti risorse:

- AWS Identity and Access Management Ruolo e politiche (IAM)
  - SystemsManager- AutomationExecutionRole concede all'utente le autorizzazioni per eseguire automazioni negli account di destinazione.
  - SystemsManager- AutomationAdministrationRole concede all'utente le autorizzazioni per eseguire automazioni in più account e unità organizzative (). OUs
- File compressi e manifest.json per il pacchetto
  - In Systems Manager, un pacchetto include almeno un file.zip di software o risorse installabili.
  - Il manifest JSON include puntatori ai file di codice del pacchetto.
- Bucket S3
  - Il pacchetto distribuito condiviso all'interno dell'organizzazione viene archiviato in modo sicuro in un bucket Amazon S3.
- AWS Systems Manager documenti (documenti SSM)
  - `DistributeSoftwarePackage`contiene la logica per distribuire il pacchetto software a ogni istanza di destinazione negli account dei membri.
  - `AddSoftwarePackageToDistributor`contiene la logica per impacchettare le risorse software installabili e aggiungerle a Automation, una funzionalità di AWS Systems Manager.
- Associazione di Systems Manager
  - Per distribuire la soluzione viene utilizzata un'associazione Systems Manager.

Architettura e flusso di lavoro

Il diagramma illustra i passaggi seguenti:

1. Per eseguire la soluzione da un account centralizzato, devi caricare i pacchetti o il software insieme alle fasi di distribuzione in un bucket S3.
2. Il pacchetto personalizzato diventa disponibile nella sezione Documenti della console Systems Manager, nella scheda Owned by me.
3. State Manager, una funzionalità di Systems Manager, crea, pianifica ed esegue un'associazione per il pacchetto all'interno dell'organizzazione. L'associazione specifica che il pacchetto software deve essere installato ed eseguito su un nodo gestito prima di poter essere installato sul nodo di destinazione.
4. L'associazione ordina a Systems Manager di installare il pacchetto sul nodo di destinazione.
5. Per eventuali installazioni o modifiche successive, gli utenti possono eseguire la stessa associazione periodicamente o manualmente da un'unica posizione per eseguire distribuzioni su più account.
6. Negli account dei membri, Automation invia i comandi di distribuzione a Distributor.
7. Il distributore distribuisce pacchetti software tra le istanze.

Questa soluzione utilizza l'account di gestione interno AWS Organizations, ma è anche possibile designare un account (amministratore delegato) per gestirlo per conto dell'organizzazione.

Tools (Strumenti)

Servizi AWS

- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati. Questo modello utilizza Amazon S3 per centralizzare e archiviare in modo sicuro il pacchetto distribuito.
- AWS Systems Managerti aiuta a gestire le applicazioni e l'infrastruttura in esecuzione in. Cloud AWS Semplifica la gestione delle applicazioni e delle risorse, riduce i tempi di rilevamento e risoluzione dei problemi operativi e aiuta a gestire le AWS risorse in modo sicuro su larga scala. Questo modello utilizza le seguenti funzionalità di Systems Manager:
  - Distributor consente di creare pacchetti e pubblicare software su istanze gestite da Systems Manager.

  - L'automazione semplifica le attività comuni di manutenzione, implementazione e riparazione per molti servizi. AWS
  - Documents esegue azioni sulle istanze gestite da Systems Manager all'interno dell'organizzazione e degli account.
- AWS Organizationsè un servizio di gestione degli account che consente di consolidare più AWS account in un'organizzazione da creare e gestire centralmente.

Altri strumenti

- Terraform è uno strumento di infrastruttura come codice (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.

Archivio di codici

Le istruzioni e il codice per questo modello sono disponibili nell'archivio GitHub centralizzato per la distribuzione dei pacchetti.

Best practice

- Per assegnare tag a un'associazione, usa il AWS Command Line Interface (AWS CLI) o il. AWS Strumenti per PowerShell Aggiunta di tag a un'associazione utilizzando la console Systems Manager non è supportata. Per ulteriori informazioni, vedere Tagging Systems Manager alle risorse nella documentazione di Systems Manager.
- Per eseguire un'associazione utilizzando una nuova versione di un documento condiviso da un altro account, imposta la versione del documento su. `default`
- Per etichettare solo il nodo di destinazione, usa una chiave tag. Se vuoi indirizzare i tuoi nodi utilizzando più chiavi di tag, usa l'opzione resource group.

Epiche

Configura i file sorgente e gli account

Operazione	Description	Competenze richieste

Clonare il repository.	1. Clona l'archivio GitHub centralizzato per la distribuzione dei pacchetti:
	`git clone https://github.com/aws-samples/aws-organization-centralised-package-distribution`
	2. Il repository di codice Terraform richiede due cartelle di personalizzazione gestite da AFT. Conferma che la tua copia locale del repository contenga queste cartelle:
	`$ cd centralised-package-distribution`
	`$ ls`
	`global-customization account-customization`
		DevOps ingegnere

Aggiorna le variabili globali.	Aggiorna i seguenti parametri di input nel `global-customization/variables.tf` file. Queste variabili si applicano a tutti gli account creati e gestiti da AFT.	DevOps ingegnere

Operazione	Description	Competenze richieste
	• `account_id` : L'ID dell'account in cui verrà implementata la soluzione Distributor. • `aws_region` : Il Regione AWS luogo in cui verrà distribuita l'associazione.	
Aggiorna le variabili dell'account.	Aggiorna i seguenti parametri di input nel `account-customization/variables.tf` file. Queste variabili si applicano solo a conti specifici creati e gestiti da AFT. • `package_bucket_name` : il nome del bucket S3 che contiene il file di distribuzione del pacchetto. • `package_name` : il nome del file di distribuzione del pacchetto. • `package_version` : la versione del pacchetto del programma di installazione.	DevOps ingegnere

Personalizza parametri e file di distribuzione

Operazione	Description	Competenze richieste
Aggiorna i parametri di input per l'associazione State Manager.	Aggiorna i seguenti parametri di input nel `account-customization/assoc`	DevOps ingegnere

Operazione	Description	Competenze richieste

	`iation.tf` file per definire lo stato che desideri mantenere sulle tue istanze. Puoi utilizzare i valori dei parametri predefiniti se supportano il tuo caso d'uso.
	• `targetAccounts` : L'unità organizzativa (OU) IDs all'interno di AWS Organizations che rappresenta gli account con le istanze di destinazione per la distribuzione. L'unità IDs organizzativa inizia con «ou».
	• `targetRegions` : Il Regioni AWS (ad esempio, «us-east-1″ o «ap-southeast-2") dove sono in esecuzione le istanze di destinazione.
	• `action`: Specificare se installare o disinstallare il pacchetto.
	• `installationType` : Uno dei seguenti tipi di installazione:
	  • `uninstall` : il pacchetto è stato disinstallato.
	  • `reinstall` : l'applicazione viene messa offline fino al completamento del processo di reinstallazione.	

Operazione	Description	Competenze richieste
	• `In-place update`: l'applicazione è disponibile mentre vengono aggiunti file nuovi o aggiornati all'installazione. • `name`: nome del pacchetto da installare o disinstallare. • `version`: la versione del pacchetto da installare o disinstallare. Se non è installata alcuna versione del pacchetto, il sistema restituisce un errore. • `bucketName` : il nome del bucket S3 in cui è stato distribuito il pacchetto. Questo bucket deve essere composto solo dai pacchetti e dal file manifest. • `bucketPrefix` : il prefisso S3 in cui sono archiviate le risorse del pacchetto. • `AutomationAssumeRole` : Amazon Resource Name (ARN) di. `SystemsManager-AutomationAdministrationRole`	

Operazione	Description	Competenze richieste
Prepara i file compressi e il `manifest.json` file per il pacchetto.	Questo modello fornisce esempi di file PowerShell installabili (.msi per Windows e.rpm per Linux) con script di installazione e disinstallazione nella cartella. `account-customization/package` 1. Sostituisci i file PowerShell installabili con i tuoi file oppure fornisci il file installabile, gli script di installazione e disinstallazione e il file manifest per creare un pacchetto nella cartella del tuo account. `account-customization` 2. Personalizza il `manifest.json` file predefinito che Terraform genera nella `account-customization` cartella in base alle tue esigenze.	DevOps ingegnere

Esegui i comandi Terraform per fornire risorse

Operazione	Description	Competenze richieste
Inizializza la configurazione Terraform.	Per implementare automaticamente la soluzione con AFT, invia il codice a: AWS CodeCommit	DevOps ingegnere

Operazione	Description	Competenze richieste

	`$ git add *`
	`$ git commit -m "message"`
	`$ git push`
	
	È inoltre possibile distribuire questa soluzione senza utilizzare AFT eseguendo un comando Terraform dalla cartella. `account-customization` Per inizializzare la directory di lavoro che contiene i file Terraform, esegui:
	
	`$ terraform init`	

Visualizza in anteprima le modifiche.	Per visualizzare in anteprima le modifiche che Terraform apporterà all'infrastruttura, esegui il comando: `$ terraform plan` Questo comando valuta la configurazione di Terraform per determinare lo stato desiderato delle risorse che sono state dichiarate. Inoltre, confronta lo stato desiderato con l'infrastruttura effettiva da fornire all'interno dell'area di lavoro.	DevOps ingegnere

Operazione	Description	Competenze richieste
Applica modifiche.	Eseguite il comando seguente per implementare le modifiche apportate ai `variables.tf` file: `$ terraform apply`	DevOps ingegnere

Convalida le risorse

Operazione	Description	Competenze richieste
Convalida la creazione di documenti SSM.	1. Nella console Systems Manager, nel riquadro di navigazione a sinistra, scegli Documenti. 2. Scegliere la scheda Owned by me (Di mia proprieta). Dovresti vedere i `AddSoftwarePackageToDistributor` pacchetti `DistributeSoftwarePackage` e.	DevOps ingegnere
Convalida la corretta implementazione delle automazioni.	1. Nella console Systems Manager, nel riquadro di navigazione a sinistra, scegli Automazione. 2. Nell'elenco delle esecuzioni di automazione, dovresti vedere le `AddSoftwarePackageToDistributor` implementazioni `DistributeSoftware`	DevOps ingegnere

Operazione	Description	Competenze richieste
	`Package` e le distribuzioni più recenti. 3. Scegli Execution ID per verificare che siano state completate correttamente.	
Verifica che il pacchetto sia stato distribuito nelle istanze di account membro interessate.	1. Nella console Systems Manager, nel riquadro di navigazione, scegli Esegui comando. 2. Nella cronologia dei comandi, vedrai ogni invocazione e il relativo stato. 3. Scegli un ID di comando per visualizzare la cronologia di distribuzione per ogni istanza di destinazione. 4. Scegli l'ID dell'istanza e controlla la sezione Output per la distribuzione.	DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
L'associazione State Manager è fallita o è bloccata in sospeso.	Consulta le informazioni sulla risoluzione dei problemi nel AWS Knowledge Center.
Impossibile eseguire un'associazione pianificata.	Le specifiche della pianificazione potrebbero non essere valide. State Manager attualmente non supporta la specificazione dei mesi nelle espressioni cron per le associazioni. Usa

Problema	Soluzione
	le espressioni cron o rate per confermare la pianificazione.

Risorse correlate

- Distribuzione centralizzata dei pacchetti (repository) GitHub
- Account Factory per Terraform (AFT)
- Casi d'uso e migliori pratiche (AWS Systems Manager documentazione)

Configura la registrazione per le applicazioni.NET in Amazon CloudWatch Logs utilizzando NLog

Bibhuti Sahu e Rob Hill (AWS), Amazon Web Services

Riepilogo

Questo modello descrive come utilizzare il framework di registrazione NLog open source per registrare l'utilizzo e gli eventi delle applicazioni.NET in Amazon CloudWatch Logs. Nella CloudWatch console, puoi visualizzare i messaggi di registro dell'applicazione quasi in tempo reale. Puoi anche impostare metriche e configurare allarmi per avvisarti se viene superata una soglia metrica. Utilizzando CloudWatch Application Insights, è possibile visualizzare dashboard automatiche o personalizzate che mostrano potenziali problemi per le applicazioni monitorate. CloudWatch Application Insights è progettato per aiutarvi a isolare rapidamente i problemi in corso con le applicazioni e l'infrastruttura.

Per scrivere messaggi di log in CloudWatch Logs, aggiungi il `AWS.Logger.NLog` NuGet pacchetto al progetto.NET. Quindi, si aggiorna il `NLog.config` file per utilizzare CloudWatch Logs come destinazione.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un'applicazione web.NET o console che:
  - Utilizza le versioni supportate di .NET Framework o.NET Core. Per ulteriori informazioni, consulta Versioni del prodotto.
  - Viene utilizzato NLog per inviare dati di registro ad Application Insights.
- Autorizzazioni per creare un ruolo IAM per un servizio AWS. Per ulteriori informazioni, consulta Autorizzazioni dei ruoli di servizio.
- Autorizzazioni per trasferire un ruolo a un servizio AWS. Per ulteriori informazioni, consulta Concessione di autorizzazioni utente per il passaggio di un ruolo a un servizio AWS.

Versioni del prodotto

- .NET Framework versione 3.5 o successiva

- .NET Core versioni 1.0.1, 2.0.0 o successive

Architecture

Stack tecnologico Target

- NLog
- CloudWatch Registri Amazon

Architettura Target

1. L'applicazione.NET scrive i dati di registro nel framework di NLog registrazione.
2. NLog scrive i dati di registro in CloudWatch Logs.
3. Si utilizzano CloudWatch allarmi e dashboard personalizzati per monitorare l'applicazione.NET.

Tools (Strumenti)

Servizi AWS

- Amazon CloudWatch Application Insights ti aiuta a osservare lo stato delle tue applicazioni e delle risorse AWS sottostanti.
- Amazon CloudWatch Logs ti aiuta a centralizzare i log di tutti i tuoi sistemi, applicazioni e servizi AWS in modo da poterli monitorare e archiviare in modo sicuro.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Tools for PowerShell è un set di PowerShell moduli che ti aiutano a creare script di operazioni sulle tue risorse AWS dalla PowerShell riga di comando.

Altri strumenti

- Logger. NLogè una NLog destinazione che registra i dati di registro in CloudWatch Logs.
- NLogè un framework di registrazione open source per piattaforme .NET che consente di scrivere dati di registro su destinazioni, come database, file di registro o console.

- PowerShellè un programma di gestione dell'automazione e della configurazione di Microsoft che funziona su Windows, Linux e macOS.
- Visual Studio è un ambiente di sviluppo integrato (IDE) che include compilatori, strumenti di completamento del codice, progettisti grafici e altre funzionalità che supportano lo sviluppo del software.

Best practice

- Imposta una politica di conservazione per il gruppo di log di destinazione. Questa operazione deve essere eseguita al di fuori della NLog configurazione. Per impostazione predefinita, i dati di registro vengono archiviati nei CloudWatch registri a tempo indeterminato.
- Aderisci alle migliori pratiche per la gestione delle chiavi di accesso AWS.

Epiche

Configura l'accesso e gli strumenti

Operazione	Description	Competenze richieste
Creare una policy IAM	Segui le istruzioni in Creazione di politiche utilizzando l'editor JSON nella documentazione IAM. Inserisci la seguente policy JSON, che dispone dei privilegi minimi necessari per consentire CloudWatch ai log di leggere e scrivere i log. `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [`	Amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste

	<pre>"logs:CreateLogGro
	up",
	
	"logs:CreateLogStr
	eam",
	
	"logs:GetLogEvents",
	
	"logs:PutLogEvents",
	
	"logs:DescribeLogG
	roups",
	
	"logs:DescribeLogS
	treams",
	
	"logs:PutRetention
	Policy"
	],
	"Resource":
	[
	"*"
	]
	}
	]
	}</pre>	

Crea un ruolo IAM.	Segui le istruzioni in Creazione di un ruolo per delegare le autorizzazioni a un servizio AWS nella documentazione IAM. Seleziona la policy che hai creato in precedenza. Questo è il ruolo che CloudWatch Logs assume per eseguire le azioni di registrazione.	Amministratore AWS, AWS DevOps

Operazione	Description	Competenze richieste
Configura AWS Tools per PowerShell.	1. Segui le istruzioni per il tuo sistema operativo in Installazione degli strumenti AWS per PowerShell. 2. Utilizza AWS Tools for PowerShell cmdlet per archiviare la chiave di accesso e la chiave segreta in un profilo. Per istruzioni, consulta Gestione dei profili negli strumenti AWS per la PowerShell documenta zione.	Informazioni generali su AWS

Configurazione NLog

Operazione	Description	Competenze richieste
Installa il NuGet pacchetto.	1. In Visual Studio, scegli File, quindi scegli Apri un progetto o una soluzione. 2. Scegli il progetto in cui desideri installarlo NLog. 3. In Visual Studio, scegli Tools, NuGet Package Manager, Package Manager Console. 4. Installa il `AWS.Logge r.NLog` NuGet pacchetto inserendo il seguente comando.	Sviluppatore di app

Operazione	Description	Competenze richieste
	`Install-Package AWS.Logger.NLog -Version 3.1.0`	

Operazione	Description	Competenze richieste
Configura la destinazione di registrazione.	1. Apri il file `NLog.config` . 2. Per il bersagliotype, inserisci`AWSTarget` . 3. Per la destinazi onelogGroup, inserisci il nome del gruppo di log che desideri utilizzare. Se il gruppo di log non esiste già, viene creato automatic amente un nuovo gruppo di log con il nome fornito. 4. Per il targetregion, inserisci la regione AWS in cui è configurato CloudWatc h Logs. 5. Per la destinazi oneprofile, inserisci il nome del profilo che hai creato in precedenz a per archiviare la chiave di accesso e la chiave segreta. 6. Salvare e chiudere il file `NLog.config` . Per un file di configurazione di esempio, consultate la sezione Informazioni aggiuntiv e di questo modello. Quando esegui l'applicazione, NLog scriverà i messaggi di registro e li invierà a CloudWatch Logs.	Sviluppatore di app

Convalida e monitora i log

Operazione	Description	Competenze richieste
Convalida la registrazione.	Segui le istruzioni in Visualizza i dati di registro inviati ai CloudWatch registri nella documentazione dei registri. CloudWatch Verifica che gli eventi di registro vengano registrati per l'applicazione.NET. Se gli eventi di registro non vengono registrati, consultate la sezione Risoluzione dei problemi in questo schema.	Informazioni generali su AWS
Monitora lo stack di applicazioni.NET.	Configura il monitoraggio in base CloudWatch alle esigenze del tuo caso d'uso. Puoi utilizzare CloudWatch Logs Insights, CloudWatch Metrics Insights e CloudWatch Application Insights per monitorare il tuo carico di lavoro.NET. Puoi anche configurare gli allarmi in modo da poterli ricevere e creare una dashboard personalizzata per monitorare il carico di lavoro da un'unica vista.	Informazioni generali su AWS

risoluzione dei problemi

Problema	Soluzione
I dati di registro non vengono visualizzati in CloudWatch Logs.	Assicurati che la policy IAM sia associata al ruolo IAM assunto da CloudWatch Logs. Per istruzioni, consulta la sezione Configurazione dell'accesso e degli strumenti nella sezione Epics.

Risorse correlate

- Lavorare con gruppi di log e flussi di log (documentazione relativa ai CloudWatch log)
- Amazon CloudWatch Logs e.NET Logging Frameworks (post sul blog AWS)

Informazioni aggiuntive

Di seguito è riportato un file di esempio. `NLog.config`

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <configSections>
    <section name="nlog" type="NLog.Config.ConfigSectionHandler, NLog" />
  </configSections>
  <startup>
    <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.7.2" />
  </startup>
  <nlog>
    <extensions>
      <add assembly="NLog.AWS.Logger" />
    </extensions>
    <targets>
      <target name="aws" type="AWSTarget" logGroup="NLog.TestGroup" region="us-east-1"
 profile="demo"/>
    </targets>
    <rules>
      <logger name="*" minlevel="Info" writeTo="aws" />
    </rules>
  </nlog>
```

```
</configuration>
```

Copia i prodotti AWS Service Catalog su diversi account AWS e regioni AWS

Sachin Vighe e Santosh Kale, Amazon Web Services

Riepilogo

AWS Service Catalog è un servizio regionale e ciò significa che i portafogli e i prodotti di AWS Service Catalog sono visibili solo nella regione AWS in cui sono stati creati. Se configuri un hub AWS Service Catalog in una nuova regione, devi ricreare i prodotti esistenti e questo processo può richiedere molto tempo.

L'approccio di questo modello aiuta a semplificare questo processo descrivendo come copiare i prodotti da un hub AWS Service Catalog in un account o in una regione AWS di origine a un nuovo hub in un account o in una regione di destinazione. Per ulteriori informazioni sul modello hub and spoke di AWS Service Catalog, consulta il modello hub and spoke di AWS Service Catalog: come automatizzare la distribuzione e la gestione di AWS Service Catalog su molti account sul blog di AWS Management and Governance.

Il modello fornisce anche i pacchetti di codice separati necessari per copiare i prodotti AWS Service Catalog tra account o in altre regioni. Utilizzando questo modello, l'organizzazione può risparmiare tempo, rendere disponibili le versioni esistenti e precedenti del prodotto in un nuovo hub AWS Service Catalog, ridurre al minimo il rischio di errori manuali e scalare l'approccio su più account o regioni.

> **Note**
>
> La sezione Epics di questo pattern offre due opzioni per copiare i prodotti. Puoi utilizzare l'Opzione 1 per copiare i prodotti tra gli account o scegliere l'Opzione 2 per copiare i prodotti tra le regioni.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Prodotti AWS Service Catalog esistenti in un account o in una regione di origine.
- Un hub AWS Service Catalog esistente in un account o in una regione di destinazione.

- Se desideri copiare prodotti tra account, devi condividere e quindi importare il portafoglio AWS Service Catalog contenente i prodotti nell'account di destinazione. Per ulteriori informazioni a riguardo, consulta Condivisione e importazione di portafogli nella documentazione di AWS Service Catalog.

Limitazioni

- I prodotti AWS Service Catalog che desideri copiare tra regioni o account non possono appartenere a più di un portafoglio.

Architecture

Il diagramma seguente mostra la copia dei prodotti AWS Service Catalog da un account di origine a un account di destinazione.

Il diagramma seguente mostra la copia dei prodotti AWS Service Catalog da una regione di origine a una regione di destinazione.

Stack tecnologico

- Amazon CloudWatch
- AWS Identity and Access Management (IAM)
- AWS Lambda
- AWS Service Catalog

Automazione e scalabilità

Puoi scalare l'approccio di questo modello utilizzando una funzione Lambda che può essere ridimensionata in base al numero di richieste ricevute o al numero di prodotti AWS Service Catalog da copiare. Per ulteriori informazioni su questo argomento, consulta la scalabilità delle funzioni Lambda nella documentazione di AWS Lambda.

Tools (Strumenti)

- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.

- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- AWS Service Catalog ti aiuta a gestire centralmente i cataloghi di servizi IT approvati per AWS. Gli utenti finali possono distribuire rapidamente soltanto i servizi IT approvati di cui hanno bisogno, in accordo con i vincoli stabiliti dall'organizzazione.

Codice

Puoi utilizzare il `cross-account-copy` pacchetto (allegato) per copiare i prodotti AWS Service Catalog tra gli account o il `cross-region-copy` pacchetto (allegato) per copiare i prodotti tra le regioni.

Il `cross-account-copy` pacchetto contiene i seguenti file:

- `copyconf.properties`— Il file di configurazione che contiene i parametri Region e AWS Account ID per copiare i prodotti tra gli account.
- `scProductCopyLambda.py`— La funzione Python per copiare prodotti tra account.
- `createDestAccountRole.sh`— Lo script per creare un ruolo IAM nell'account di destinazione.
- `createSrcAccountRole.sh`— Lo script per creare un ruolo IAM nell'account di origine.
- `copyProduct.sh`— Lo script per creare e richiamare la funzione Lambda per copiare i prodotti tra account.

Il `cross-region-copy` pacchetto contiene i seguenti file:

- `copyconf.properties`— Il file di configurazione che contiene i parametri Region e ID dell'account AWS per copiare i prodotti tra le regioni.
- `scProductCopyLambda.py`— La funzione Python per copiare prodotti tra regioni.
- `copyProduct.sh`— Lo script per creare un ruolo IAM e creare e richiamare la funzione Lambda per copiare i prodotti tra le regioni.

Epiche

Opzione 1: copia i prodotti AWS Service Catalog su più account

Operazione	Description	Competenze richieste
Aggiorna il file di configurazione.	1. Scarica il `cross-account-copy` pacchetto (allegato) sul tuo computer locale. 2. Aggiorna il file di `copyconf.properties` configurazione con i seguenti valori: • `srcRegion` — Fornisci la regione di origine che contiene i prodotti. • `destRegion` — Fornisci la regione di destinazione per i prodotti. • `sourceAccountId` — Fornisci l'ID dell'account AWS per il tuo account di origine. • `destAccountId` — Fornisci l'ID dell'account AWS per l'account di destinazione.	Amministratore AWS, amministratore di sistema AWS, amministratore cloud
Configura le tue credenziali per AWS CLI nell'account di destinazione.	Configura le tue credenziali per accedere a AWS CLI nel tuo account di destinazione eseguendo `aws configure` il comando e fornendo i seguenti valori:	Amministratore AWS, amministratore di sistema AWS, amministratore cloud

Operazione	Description	Competenze richieste

	`$aws configure`
	`AWS Access Key ID`
	` [None]: <your_acc`
	`ess_key_id>`
	`AWS Secret Access Key`
	` [None]: <your_sec`
	`ret_access_key>`
	`Default region name`
	` [None]: Region`
	`Default output format`
	` [None]:`
	
	Per ulteriori informazioni su questo argomento, consulta le nozioni di base sulla configurazione nella documentazione dell'interfaccia a riga di comando AWS.	

Operazione	Description	Competenze richieste

Configura le tue credenziali per AWS CLI nell'account di origine.	Configura le tue credenziali per accedere alla CLI di AWS nel tuo account di origine eseguendo `aws configure` il comando e fornendo i seguenti valori:
	`$aws configure`
	`AWS Access Key ID [None]: <your_access_key_id>`
	`AWS Secret Access Key [None]: <your_secret_access_key>`
	`Default region name [None]: Region`
	`Default output format [None]:`
	Per ulteriori informazioni su questo argomento, consulta le nozioni di base sulla configurazione nella documentazione dell'interfaccia a riga di comando AWS.
		Amministratore AWS, amministratore di sistema AWS, amministratore cloud

Operazione	Description	Competenze richieste
Crea un ruolo di esecuzione Lambda nel tuo account di destinazione.	Esegui lo `createDestAccountRole.sh` script nell'account di destinazione. Lo script implementa le seguenti azioni: • Crea un ruolo di esecuzione Lambda nell'account di destinazione • Crea e allega la policy IAM per il ruolo di esecuzione Lambda	Amministratore AWS, amministratore di sistema AWS, amministratore cloud
Crea il ruolo IAM tra account diversi nel tuo account di origine.	Esegui lo `createSrcAccountRole.sh` script nel tuo account di origine. Lo script implementa le seguenti azioni: • Crea un ruolo IAM tra account nell'account di origine che viene assunto dal ruolo di esecuzione Lambda nell'account di destinazione per copiare i prodotti • Crea e allega una policy IAM per il ruolo tra account diversi nell'account di origine	Amministratore AWS, amministratore di sistema AWS, amministratore cloud

Operazione	Description	Competenze richieste
Esegui lo script CopyProduct nell'account di destinazione.	Esegui lo `copyProduct.sh` script nell'account di destinazione. Lo script implementa le seguenti azioni: • Crea e richiama la funzione Lambda per copiare i prodotti dall'account di origine all'account di destinazione	Amministratore AWS, amministratore di sistema AWS, amministratore cloud

Opzione 2: copiare i prodotti AWS Service Catalog da una regione di origine a una regione di destinazione

Operazione	Description	Competenze richieste
Aggiorna il file di configurazione.	1. Scarica il `cross-region-copy` pacchetto (allegato) sul tuo computer locale. 2. Aggiorna il file di `copyconf.properties` configurazione con i seguenti valori: • `srcRegion` — Fornisci la regione di origine che contiene i prodotti. • `destRegion` — Fornisci la regione di destinazione per i prodotti. • `accountId` — Fornisci l'ID del tuo account AWS.	Amministratore di sistema AWS, amministratore cloud, amministratore AWS

Operazione	Description	Competenze richieste

Configura le tue credenziali per AWS CLI.	Configura le tue credenziali per accedere a AWS CLI nel tuo ambiente eseguendo `aws configure` il comando e fornendo i seguenti valori:
	`$aws configure`
	`AWS Access Key ID [None]: <your_access_key_id>`
	`AWS Secret Access Key [None]: <your_secret_access_key>`
	`Default region name [None]: Region`
	`Default output format [None]:`
	Per ulteriori informazioni su questo argomento, consulta le nozioni di base sulla configurazione nella documentazione dell'interfaccia a riga di comando AWS.
		Amministratore AWS, amministratore di sistema AWS, amministratore cloud

Operazione	Description	Competenze richieste
Esegui lo script CopyProduct.	Esegui lo `copyProduct.sh` script nella regione di destinazione. Lo script implementa le seguenti azioni: • Crea un ruolo di esecuzione Lambda • Crea e allega la policy IAM per il ruolo di esecuzione Lambda • Crea e richiama la funzione Lambda per copiare i prodotti dalla regione di origine alla regione di destinazione	Amministratore AWS, amministratore di sistema AWS, amministratore cloud

Risorse correlate

- Creare un ruolo di esecuzione Lambda (documentazione AWS Lambda)
- Creare una funzione Lambda (documentazione AWS Lambda)
- Riferimento all'API AWS Service Catalog
- Documentazione di AWS Service Catalog

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Crea una matrice RACI o RASCI per un modello operativo cloud

Teddy Germade, Jerome Descreux, Florian Leroux e Josselin LE MINEUR, Amazon Web Services

Riepilogo

Il Cloud Center of Excellence (CCoE) o CEE (Cloud Enablement Engine) è un team competente e responsabile che si concentra sulla preparazione operativa per il cloud. Il loro obiettivo principale è trasformare l'organizzazione IT delle informazioni da un modello operativo locale a un modello operativo cloud. L' CCoE dovrebbe essere un team interfunzionale che includa la rappresentanza dell'infrastruttura, delle applicazioni, delle operazioni e della sicurezza.

Uno dei componenti chiave di un modello operativo cloud è una matrice RACI o una matrice RASCI. Viene utilizzato per definire i ruoli e le responsabilità di tutte le parti coinvolte nelle attività di migrazione e nelle operazioni cloud. Il nome della matrice deriva dai tipi di responsabilità definiti nella matrice: responsabile (R), responsabile (A), supporto (S), consultato (C) e informato (I). Il tipo di supporto è facoltativo. Se la includi, viene chiamata matrice RASCI e se la escludi, viene chiamata matrice RACI.

Partendo dal modello allegato, il team CCo E può creare una matrice RACI o RASCI per l'organizzazione. Il modello contiene team, ruoli e attività comuni nei modelli operativi cloud. La base di questa matrice sono le attività relative all'integrazione delle operazioni e alle funzionalità CCo E. Tuttavia, è possibile personalizzare questo modello per soddisfare le esigenze della struttura e del caso d'uso dell'organizzazione.

Non ci sono limiti all'implementazione di una matrice RACI. Questo approccio funziona per le grandi organizzazioni, le start-up e tutto il resto. Per le piccole organizzazioni, la stessa risorsa può ricoprire diversi ruoli.

Epiche

Crea la matrice

Operazione	Description	Competenze richieste
Identifica le principali parti interessate.	Identifica i responsabili chiave dei servizi e dei team collegati agli obiettivi strategici del tuo modello operativo cloud.	Project manager

Operazione	Description	Competenze richieste

Personalizza il modello di matrice.	Scaricate il modello nella sezione Allegati, quindi aggiornate la matrice RACI o RASCI come segue:
	• Nel foglio di lavoro Cloud Teams, aggiorna i nomi degli CCo E stream, i nomi dei team e le descrizioni dei team in base alle esigenze della tua organizzazione.
	• Nel foglio di lavoro Cloud Roles, aggiorna i ruoli, i nomi dei team e le descrizio ni dei ruoli in base alle esigenze della tua organizza zione.
	• Nel foglio di lavoro RASCI, aggiorna quanto segue in base alle esigenze della tua organizzazione:
	  • Nella riga 1 e nella colonna A, aggiorna i flussi CCo E.
	  • Nella riga 2, aggiorna i nomi delle squadre.
	  • Nella riga 3, aggiorna i nomi dei ruoli.
	  • Nelle colonne D ed E, aggiorna i campi e le attività generali che desideri includere nel grafico RASCI.
		Project manager

Operazione	Description	Competenze richieste
Pianifica le riunioni.	1. Comunica gli obiettivi RASCI a tutte le parti interessate. 2. Pianifica una o più riunioni in modo che possa partecipare un rappresen tante autorizzato di ogni team.	Project manager

Operazione	Description	Competenze richieste
Completa la matrice.	Durante la riunione con tutte le parti interessate, procedi come segue: 1. Conferma la presenza di un rappresentante di ogni team. La partecipazione del team è obbligatoria in modo da poter assegnare con precisione i tipi di responsabilità per ogni attività. 2. Rivedi cos'è una matrice RASCI e gli obiettivi con i partecipanti. 3. Esamina il modello di responsabilità condivisa con i partecipanti in modo che comprendano la portata delle responsabilità della loro organizzazione per la sicurezza nel cloud. 4. Nel foglio di lavoro RASCI, per ogni attività o attività, completa le colonne da F a AN per assegnare i seguenti tipi di responsab ilità: • Responsabile (R): questo ruolo è responsabile dell'esecuzione del lavoro necessario per completar e l'attività.	Project manager

Operazione	Description	Competenze richieste
	• Responsabile (A): questo ruolo ha la responsabilità di garantire il completamento dell'attività. Questo ruolo è anche responsabile di garantire che i prerequisiti siano soddisfatti e di delegare l'attività ai responsabili. • Support (S): questo ruolo aiuta i responsabili a completare l'attività. Questo tipo di responsabilità è facoltativo e puoi scegliere di escluderlo per creare una matrice RACI più tradizionale. • Consultato (C): questo ruolo dovrebbe essere consultato per opinioni o competenze in merito all'attività. A seconda dell'attività, questo tipo di responsabilità potrebbe non essere richiesto. • Informato (I): questo ruolo deve essere tenuto aggiornato sullo stato di avanzamento dell'attività e avvisato quando l'attività viene completata.	

Operazione	Description	Competenze richieste
	• Vuoto: questo ruolo non è coinvolto nell'attività o nell'attività.	
Condividi la matrice RASCI.	Quando la matrice RACI o RASCI è completa, fatela approvare dalla dirigenza . Salvalo in un archivio condiviso o in una posizione centrale in cui tutte le parti interessate possano accedervi . Si consiglia di utilizzare processi standard di controllo dei documenti per registrare e approvare le revisioni della matrice.	Project manager

Risorse correlate

- Modello di responsabilità condivisa AWS

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Crea allarmi per metriche personalizzate utilizzando il rilevamento delle anomalie di Amazon CloudWatch

Ram Kandaswamy e Raheem Jiwani, Amazon Web Services

Riepilogo

Sul cloud Amazon Web Services (AWS), puoi utilizzare Amazon CloudWatch per creare allarmi che monitorano i parametri e inviano notifiche o apportano automaticamente modifiche in caso di superamento di una soglia.

Per evitare di essere limitati da soglie statiche, puoi creare allarmi basati su modelli passati che ti avvisino se determinati parametri non rientrano nella normale finestra operativa. Ad esempio, puoi monitorare i tempi di risposta della tua API da Amazon API Gateway e ricevere notifiche sulle anomalie che ti impediscono di rispettare un accordo sul livello di servizio (SLA).

Questo modello descrive come utilizzare il rilevamento delle CloudWatch anomalie per metriche personalizzate. Il modello mostra come creare una metrica personalizzata in Amazon CloudWatch Logs Insights o pubblicare una metrica personalizzata con una funzione AWS Lambda, quindi configurare il rilevamento delle anomalie e creare notifiche utilizzando Amazon Simple Notification Service (Amazon SNS).

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un argomento SNS esistente, configurato per inviare notifiche e-mail. Per ulteriori informazioni su questo argomento, consulta la sezione Guida introduttiva ad Amazon SNS nella documentazione di Amazon SNS.
- Un'applicazione esistente, configurata con CloudWatch Logs.

Limitazioni

- CloudWatch le metriche non supportano intervalli di tempo di millisecondi. Per ulteriori informazioni sulla granularità delle metriche regolari e personalizzate, consulta Amazon. CloudWatch FAQs

Architecture

Il diagramma mostra il flusso di lavoro seguente:

1. I log che utilizzano metriche create e aggiornate da CloudWatch Logs vengono trasmessi in streaming a. CloudWatch
2. Un allarme viene avviato in base a soglie e invia un avviso a un argomento SNS.
3. Amazon SNS ti invia una notifica via e-mail.

Stack tecnologico

- CloudWatch
- AWS Lambda
- Amazon SNS

Tools (Strumenti)

- Amazon CloudWatch offre una soluzione di monitoraggio affidabile, scalabile e flessibile.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza effettuare il provisioning o gestire server.
- Amazon Simple Notification Service (Amazon SNS) è un servizio gestito che fornisce il recapito dei messaggi dagli editori agli abbonati.

Epiche

Imposta il rilevamento delle anomalie per una metrica personalizzata

Operazione	Description	Competenze richieste
Opzione 1: crea una metrica personalizzata con una funzione Lambda.	Scarica il `lambda_function.py` file (allegato) e sostituisci il `lambda_function.py` file di esempio nel aws-lambda-developer-guiderepository su AWS	DevOps ingegnere, AWS DevOps

Operazione	Description	Competenze richieste
	Documentation GitHub. Ciò fornisce una funzione Lambda di esempio che invia metriche personalizzate ai registri. CloudWatch La funzione Lambda utilizza l'API Boto3 per l'integrazione con. CloudWatch Dopo aver eseguito la funzione Lambda, puoi accedere alla Console di gestione AWS, aprire la CloudWatch console e la metrica pubblicata è disponibi le nello spazio dei nomi pubblicato.	

Operazione	Description	Competenze richieste
Opzione 2: crea metriche personalizzate da gruppi di CloudWatch log.	Accedi alla Console di gestione AWS, apri la CloudWatch console e scegli Gruppi di log. Scegli il gruppo di log per cui vuoi creare una metrica. Scegli Azioni, quindi scegli Crea filtro metrico. In Schema di filtro, inserisci il modello di filtro che desideri utilizzar e. Per ulteriori informazioni, consultate Filter and pattern syntax nella CloudWatch documentazione. Per testare il modello di filtro, inserisci uno o più eventi di registro in Test Pattern. Ogni log eventi deve essere all'inter no di una riga, in quanto le interruzioni di riga vengono utilizzate per separare i log eventi nella casella Log event messages (Messaggi di registro eventi). Dopo aver testato il pattern, puoi inserire un nome e un valore per la metrica in Dettagli metrici. Per ulteriori informazioni e passaggi per creare una metrica personalizzata, consulta Creare un filtro metrico per un gruppo di	DevOps ingegnere, AWS DevOps

Operazione	Description	Competenze richieste
	log nella documentazione. CloudWatch	

Operazione	Description	Competenze richieste
Crea un allarme per la tua metrica personalizzata.	Sulla CloudWatch console, scegli Allarmi, quindi scegli Crea allarme. Scegli Seleziona metrica e inserisci il nome della metrica che hai creato in precedenza nella casella di ricerca. Scegli la scheda Metriche grafiche e configura le opzioni in base alle tue esigenze. In Condizioni, scegli Rilevamento delle anomalie anziché Soglie statiche. Questo mostra una banda basata su due deviazion i standard predefinite. È possibile impostare soglie e regolarle in base alle proprie esigenze. Scegli Next (Successivo). **Note** La banda è dinamica e dipende dalla qualità dei punti dati. Quando inizi ad aggregare più dati, la banda e le soglie vengono aggiornate automatic amente.	DevOps ingegnere, AWS DevOps

Operazione	Description	Competenze richieste
Configura le notifiche SNS.	In Notifica, scegli l'argomento SNS per notificare quando l'allarme è in `ALARM` stato, `OK` stato o `INSUFFICIENT_DATA` stato. Per fare in modo che l'allarme invii più notifiche per lo stesso stato di allarme o per stati di allarme diversi, scegli Add notification (Aggiungi notifica). Scegli Next (Successivo). Inserisci un nome e una descrizione per l'allarme. Il nome deve contenere solo caratteri ASCII. Quindi scegli Successivo. In Anteprima e crea, conferma che le informazioni e le condizioni siano corrette, quindi scegli Crea allarme.	DevOps ingegnere, AWS DevOps

Risorse correlate

- Pubblicazione di metriche personalizzate su CloudWatch
- Utilizzo del rilevamento CloudWatch delle anomalie
- Eventi di allarme e Amazon EventBridge
- Quali sono le migliori pratiche da seguire quando si inseriscono metriche personalizzate su Cloud Watch? (video)
- Introduzione a CloudWatch Application Insights (video)
- Rileva le anomalie con CloudWatch (video)

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Crea un IDE AWS Cloud9 che utilizza volumi Amazon EBS con crittografia predefinita

Janardhan Malyala e Dhrubajyoti Mukherjee, Amazon Web Services

Riepilogo

Avviso: non AWS Cloud9 è più disponibile per i nuovi clienti. I clienti esistenti di AWS Cloud9 possono continuare a utilizzare il servizio normalmente. Ulteriori informazioni

Puoi utilizzare la crittografia di default per applicare la crittografia dei tuoi volumi Amazon Elastic Block Store (Amazon EBS) e delle copie degli snapshot sul cloud Amazon Web Services (AWS).

Puoi creare un ambiente di sviluppo integrato (IDE) AWS Cloud9 che utilizza volumi EBS crittografati per impostazione predefinita. Tuttavia, il ruolo collegato al servizio AWS Identity and Access Management (IAM) per AWS Cloud9 richiede l'accesso alla chiave AWS Key Management Service (AWS KMS) per questi volumi EBS. Se l'accesso non viene fornito, l'IDE AWS Cloud9 potrebbe non avviarsi e il debug potrebbe essere difficile.

Questo modello fornisce i passaggi per aggiungere il ruolo collegato ai servizi per AWS Cloud9 alla chiave AWS KMS utilizzata dai volumi EBS. La configurazione descritta da questo modello ti aiuta a creare e avviare con successo un IDE che utilizza volumi EBS con crittografia per impostazione predefinita.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- La crittografia predefinita è attivata per i volumi EBS. Per ulteriori informazioni sulla crittografia predefinita, consulta la crittografia Amazon EBS nella documentazione di Amazon Elastic Compute Cloud EC2 (Amazon).
- Una chiave KMS esistente gestita dal cliente per crittografare i volumi EBS.

> **Note**
>
> Non è necessario creare il ruolo collegato ai servizi per AWS Cloud9. Quando crei un ambiente di sviluppo AWS Cloud9, AWS Cloud9 crea il ruolo collegato al servizio per te.

Architecture

Stack tecnologico

- AWS Cloud9
- IAM
- AWS KMS

Tools (Strumenti)

- AWS Cloud9 è un ambiente di sviluppo integrato (IDE) che ti aiuta a codificare, creare, eseguire, testare ed eseguire il debug del software. Ti aiuta anche a rilasciare software nel cloud AWS.
- Amazon Elastic Block Store (Amazon EBS) fornisce volumi di storage a livello di blocco da utilizzare con istanze Amazon Elastic Compute Cloud (Amazon). EC2
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.

Epiche

Trova il valore della chiave di crittografia predefinita

Operazione	Description	Competenze richieste
Registra il valore della chiave di crittografia predefinita per i volumi EBS.	Accedi alla Console di gestione AWS e apri la EC2 console Amazon. Scegli EC2 dashboard, quindi scegli Protezione e sicurezza dei dati negli attributi dell'acco unt. Nella sezione di crittogra fia EBS, copia e registra il valore nella chiave di crittogra fia predefinita.	Architetto del cloud, DevOps ingegnere

Fornisci l'accesso alla chiave AWS KMS

Operazione	Description	Competenze richieste
Fornisci ad AWS Cloud9 l'accesso alla chiave KMS per i volumi EBS.	1. Apri la console AWS KMS, quindi scegli Customer managed keys. Seleziona la chiave AWS KMS utilizzata per la crittografia Amazon EBS, quindi scegli Visualizza chiave. 2. Nella scheda Key policy, conferma di poter visualizzare il formato testuale della policy chiave. Se non riesci a visualizzare il modulo di testo, scegli Passa alla visualizzazione delle norme. 3. Scegli Modifica. Aggiungi il codice nella sezione Informazioni aggiuntive alla politica, quindi scegli Salva modifiche. Le modifiche alle policy consentono al ruolo collegato al servizio di AWS Cloud9`AWSServiceRoleForAWSCloud9` , di accedere alla chiave. Per ulteriori informazioni sull'aggiornamento di una policy chiave, consulta How to change a key policy (documentazione AWS KMS).	Architetto del cloud, ingegnere DevOps

Operazione	Description	Competenze richieste
	Important Il ruolo collegato ai servizi per AWS Cloud9 viene creato automaticamente all'avvio del primo IDE. Per ulteriori informazioni, consulta Creazione di un ruolo collegato ai servizi nella documentazione di AWS Cloud9.	

Crea e avvia l'IDE

Operazione	Description	Competenze richieste
Crea e avvia l'IDE AWS Cloud9.	Apri la console AWS Cloud9 e scegli Crea ambiente. Configura l'IDE in base alle tue esigenze seguendo i passaggi descritti nella sezione Creazione di un EC2 ambiente nella documenta zione di AWS Cloud9.	Architetto del cloud, ingegnere DevOps

Risorse correlate

- Crittografa i volumi EBS utilizzati da AWS Cloud9
- Crea un ruolo collegato ai servizi per AWS Cloud9
- Crea un EC2 ambiente in AWS Cloud9

Informazioni aggiuntive

Aggiornamenti delle policy chiave di AWS KMS

Sostituisci <aws_accountid> con il tuo ID account AWS.

```
{
            "Sid": "Allow use of the key",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::<aws_accountid>:role/aws-service-role/
cloud9.amazonaws.com/AWSServiceRoleForAWSCloud9"
            },
            "Action": [
                "kms:Encrypt",
                "kms:Decrypt",
                "kms:ReEncrypt*",
                "kms:GenerateDataKey*",
                "kms:DescribeKey"
            ],
            "Resource": "*"
        },
        {
            "Sid": "Allow attachment of persistent resources",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::<aws_accountid>:role/aws-service-role/
cloud9.amazonaws.com/AWSServiceRoleForAWSCloud9"
            },
            "Action": [
                "kms:CreateGrant",
                "kms:ListGrants",
                "kms:RevokeGrant"
            ],
            "Resource": "*",
            "Condition": {
                "Bool": {
                    "kms:GrantIsForAWSResource": "true"
                }
            }
        }
```

Utilizzo di una chiave per più account

Se desideri utilizzare una chiave KMS per più account, devi utilizzare una concessione in combinazione con la politica delle chiavi KMS. Ciò consente l'accesso alla chiave da più account. Nello stesso account che hai usato per creare l'ambiente Cloud9, esegui il seguente comando nel terminale.

```
aws kms create-grant \
 --region <Region where Cloud9 environment is created> \
 --key-id <The cross-account KMS key ARN> \
 --grantee-principal arn:aws:iam::<The account where Cloud9 environment is
 created>:role/aws-service-role/cloud9.amazonaws.com/AWSServiceRoleForAWSCloud9 \
 --operations "Encrypt" "Decrypt" "ReEncryptFrom" "ReEncryptTo" "GenerateDataKey"
 "GenerateDataKeyWithoutPlaintext" "DescribeKey" "CreateGrant"
```

Dopo aver eseguito questo comando, puoi creare ambienti Cloud9 utilizzando la crittografia EBS con una chiave in un account diverso.

Crea CloudWatch dashboard Amazon basate su tag automaticamente

Janak Vadaria, Vinodkumar Mandalapu e RAJNEESH TYAGI, Amazon Web Services

Riepilogo

La creazione manuale di diversi CloudWatch dashboard Amazon può richiedere molto tempo, in particolare quando devi creare e aggiornare più risorse per scalare automaticamente il tuo ambiente. Una soluzione che crea e aggiorna automaticamente le CloudWatch dashboard può farti risparmiare tempo. Questo modello ti aiuta a implementare una AWS Cloud Development Kit (AWS CDK) pipeline completamente automatizzata che crea e aggiorna i CloudWatch dashboard per AWS le tue risorse in base agli eventi di modifica dei tag, per visualizzare le metriche Golden Signals.

In Site Reliability Engineering (SRE), Golden Signals si riferisce a un set completo di metriche che offrono una visione ampia di un servizio dal punto di vista dell'utente o del consumatore. Queste metriche comprendono latenza, traffico, errori e saturazione. Per ulteriori informazioni, consulta Cos'è il Site Reliability Engineering (SRE)? sul AWS sito Web.

La soluzione fornita da questo modello è basata sugli eventi. Dopo l'implementazione, monitora continuamente gli eventi di modifica dei tag e aggiorna automaticamente i dashboard e gli CloudWatch allarmi.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- AWS Command Line Interface (AWS CLI), installato e configurato
- Prerequisiti per la versione v2 AWS CDK
- Un ambiente con bootstrap su AWS
- Python versione 3
- AWS SDK per Python (Boto3), installato
- Node.js versione 18 o successiva
- Node package manager (npm), installato e configurato per AWS CDK
- Familiarità moderata (livello 200) con e AWS CDK AWS CodePipeline

Limitazioni

Questa soluzione attualmente crea dashboard automatizzate solo per i seguenti servizi AWS:

- Amazon Relational Database Service (Amazon RDS)
- AWS Auto Scaling
- Amazon Simple Notification Service (Amazon SNS)
- Amazon DynamoDB
- AWS Lambda

Architecture

Stack tecnologico Target

- CloudWatch dashboard
- CloudWatch allarmi

Architettura di destinazione

1. Un evento di modifica dei AWS tag dell'applicazione configurata o le modifiche al codice avvia una pipeline AWS CodePipeline per creare e distribuire dashboard aggiornati. CloudWatch
2. AWS CodeBuild esegue uno script Python per trovare le risorse con tag configurati e memorizza la risorsa IDs in un file locale in un CodeBuild ambiente.
3. CodeBuild esegue cdk synth per generare CloudFormation modelli che implementano dashboard e allarmi. CloudWatch
4. CodePipeline distribuisce i CloudFormation modelli nella regione e nella regione specificata. Account AWS
5. Quando lo CloudFormation stack è stato distribuito correttamente, puoi visualizzare i CloudWatch dashboard e gli allarmi.

Automazione e scalabilità

Questa soluzione è stata automatizzata utilizzando il AWS CDK. Puoi trovare il codice nel CloudWatch repository GitHub Golden Signals Dashboards su Amazon. Per una scalabilità aggiuntiva e per creare dashboard personalizzate, puoi configurare più chiavi e valori di tag.

Tools (Strumenti)

Servizi Amazon

- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti, tra cui AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altro modo. Account AWS
- AWS CodePipelineti aiuta a modellare e configurare rapidamente le diverse fasi di un rilascio del software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- AWS CodeBuildè un servizio di compilazione completamente gestito che consente di compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per l'implementazione.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella shell della riga di comando.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Best practice

Come best practice di sicurezza, puoi utilizzare la crittografia e l'autenticazione per gli archivi di origine che si connettono alle tue pipeline. Per ulteriori best practice, consulta le CodePipeline best practice e i casi d'uso nella CodePipeline documentazione.

Epiche

Configura e distribuisci l'applicazione di esempio

Operazione	Description	Competenze richieste
Configura e distribuisci l'applicazione di esempio.	1. Clona il repository GitHub di codice di esempio utilizzando il comando:	AWS DevOps

Operazione	Description	Competenze richieste
	`git clone https://github.com/aws-samples/golden-signals-dashboards-sample-app` 2. Accedi al repository clonato sul tuo computer e apri il `src/project-settings.ts` file con l'editor che preferisci. 3. Modifica il valore `projectSettings` costante in base ai tag AWS delle risorse e alle mappature delle applicazioni. 4. Imposta le `AWS_ACCOUNT` variabili`AWS_REGION` , e di `GS_DASHBOARD_INSTANCE` ambiente: • `AWS_ACCOUNT` Imposta l'ID dell' AWS account. • `AWS_REGION` Imposta la regione in cui desideri distribuire l'applicazione di esempio. • Impostato `GS_DASHBOARD_INSTANCE` su`dev`, o `testprod`, a seconda dell'ambiente di sviluppo in uso. (È consigliabile utilizzare `test` la procedura di	

Operazione	Description	Competenze richieste
	test descritta in questo schema). 5. Configura il AWS CLI con le tue AWS credenzia li. Per ulteriori informazi oni, consulta Impostare e visualizzare le impostazioni di configurazione utilizzan do i comandi nella AWS CLI documentazione. 6. Esegui il comando seguente per distribuire l'applicazione di esempio del dashboard Golden Signals: `sh deploy.sh`	

Operazione	Description	Competenze richieste
Crea automaticamente dashboard e allarmi.	Dopo aver distribuito l'applicazione di esempio, puoi creare qualsiasi risorsa supportata da questa soluzione con i valori di tag previsti, che creeranno automaticamente i dashboard e gli allarmi specificati. Per testare questa soluzione, create una funzione: AWS Lambda 1. Accedi al sito Console di gestione AWS in Regione AWS cui hai distribuito l'applicazione di esempio. 2. Apri la console Lambda all'indirizzo. https://console.aws.amazon.com/lambda/ 3. Scegli Crea una funzione, quindi inserisci il nome di una funzione. 4. Nel riquadro Impostazioni avanzate, seleziona Abilita tag, quindi scegli Aggiungi nuovo tag. Inserisci la chiave e il valore seguenti: • Chiave: `AutoDashboard` • Valore: `True` 5. Scegli Crea funzione.	AWS DevOps

Operazione	Description	Competenze richieste
	La funzione Lambda avvia immediatamente una pipeline di codice, che crea automaticamente i dashboard e gli allarmi per quella particolare funzione Lambda. 6. Per visualizzare i dashboard e gli allarmi automatici, apri la console all'indirizzo. CloudWatch https://console.aws.amazon.com/cloudwatch/ Puoi visualizzare i dashboard e gli allarmi personalizzati per la funzione specificata nella `projectSettings` costante (APP1-lambda per impostazione predefinita). 7. Seleziona la dashboard per la funzione Lambda per visualizzare dashboard automatici aggiuntivi creati come parte di questa soluzione. 8. Ripeti questi passaggi per altri servizi, come Amazon RDS, Amazon SNS e DynamoDB AWS Auto Scaling, per generare i dashboard associati. Per un esempio per Amazon	

Operazione	Description	Competenze richieste
	RDS, consulta la sezione Informazioni aggiuntive.	

Rimuovi l'applicazione di esempio

Operazione	Description	Competenze richieste
Rimuovi il `golden-signals-dashboard` costrutto.	1. Per rimuovere tutti gli CloudFormation stack creati dall'applicazione di esempio, è necessario riconfigurare le variabili `AWS_ACCOUNT` `AWS_REGION` , e `GS_DASHBOARD_INSTANCE` di ambiente. Il `destroy.sh` comando richiede queste configurazioni. • `AWS_ACCOUNT` è l'ID dell'AWS account. • `AWS_REGION` è la regione in cui è stata distribuita l'applicazione di esempio. • `GS_DASHBOARD_INSTANCE` è `dev` o `test` `prod` si basa sulle impostazioni precedenti. 2. Configura AWS CLI con le tue AWS credenziali. 3. Esegui il comando seguente per rimuovere	AWS DevOps

Operazione	Description	Competenze richieste
	l'applicazione di esempio e tutti gli CloudFormation stack associati: `sh destroy.sh`	

risoluzione dei problemi

Problema	Soluzione
Comando Python non trovato (riferimento alla `findresources.sh` riga 8).	Controlla la versione della tua installazione di Python. Se hai installato Python versione 3, sostituiscilo `python` con la `python3` riga 8 del `resources.sh` file ed esegui nuovamente il `sh deploy.sh` comando per distribuire la soluzione.

Risorse correlate

- Bootstrap (documentazione)AWS CDK
- Utilizzo di profili denominati (documentazione)AWS CLI
- AWS CDK Workshop

Informazioni aggiuntive

L'illustrazione seguente mostra un dashboard di esempio per Amazon RDS creato come parte di questa soluzione.

Documenta il progetto della tua landing zone AWS

Michael Daehnert, Florian Langer e Michael Lodemann, Amazon Web Services

Riepilogo

Una landing zone è un ambiente multi-account ben progettato, basato sulle migliori pratiche di sicurezza e conformità. È il contenitore a livello aziendale che contiene tutte le unità organizzative (OUs) Account AWS, gli utenti e le altre risorse. Una landing zone può essere scalata per soddisfare le esigenze di un'azienda di qualsiasi dimensione. AWS ha due opzioni per creare la tua landing zone: una landing zone basata sui servizi AWS Control Towero una landing zone personalizzata che crei tu. Ogni opzione richiede un diverso livello di conoscenza. AWS

AWS creato AWS Control Tower per aiutarti a risparmiare tempo automatizzando la configurazione di una landing zone. AWS Control Tower è gestito AWS e utilizza le migliori pratiche e linee guida per aiutarti a creare il tuo ambiente di base. AWS Control Tower utilizza servizi integrati, come AWS Service Catalogue AWS Organizations, per fornire account nella landing zone dell'utente e gestire l'accesso a tali account.

AWS i progetti di landing zone variano in termini di requisiti, dettagli di implementazione e azioni operative. Ci sono aspetti di personalizzazione che devono essere gestiti con ogni implementazione di landing zone. Ciò include (ma non è limitato a) il modo in cui viene gestita la gestione degli accessi, lo stack tecnologico utilizzato e quali sono i requisiti di monitoraggio per l'eccellenza operativa. Questo modello fornisce un modello che ti aiuta a documentare il tuo progetto di landing zone. Utilizzando il modello, puoi documentare il tuo progetto più rapidamente e aiutare i team di sviluppo e operativi a comprendere la tua landing zone.

Prerequisiti e limitazioni

Limitazioni

Questo modello non descrive cos'è una landing zone o come implementarne una. Per ulteriori informazioni su questi argomenti, consulta la sezione Risorse correlate.

Epiche

Crea il documento di progettazione

Operazione	Description	Competenze richieste
Identifica le principali parti interessate.	Identifica i responsabili chiave del servizio e del team collegati alla tua landing zone.	Project manager
Personalizza il modello.	Scarica il modello nella sezione Allegati, quindi aggiorna il modello come segue: 1. Rimuovi tutte le sezioni che non si applicano alla landing zone o ai processi della tua organizzazione. 2. Aggiungi tutte le sezioni che sono uniche per la tua organizzazione.	Project manager
Completa il modello.	Nelle riunioni con le parti interessate o utilizzando un write-and-review processo, completa il modello come segue: 1. Utilizza le linee guida e le informazioni nelle caselle blu per completare ogni sezione. 2. Sostituisci o rimuovi i campi gialli con valori personali zzati per la tua organizza zione.	Project manager

Operazione	Description	Competenze richieste
	3. Sostituisci o rimuovi qualsiasi campo di immagine con la tua architettura o i tuoi diagrammi di flusso personalizzati. 4. Completa la sezione Cronologia delle revisioni e Collaboratori del modello.	
Condividi il documento di progettazione.	Quando la documentazione sulla progettazione della landing zone è completa, salvala in un archivio condiviso o in una posizione centrale in cui tutte le parti interessate possano accedervi . Si consiglia di utilizzar e processi di controllo dei documenti standard per registrare e approvare le revisioni del documento di progettazione.	Project manager

Risorse correlate

- AWS Control Tower documentazione
  - Pianifica la tua AWS Control Tower landing zone
  - AWS strategia multi-account per la tua AWS Control Tower landing zone
  - Suggerimenti amministrativi per la configurazione delle landing zone
  - Aspettative per la configurazione delle landing zone
- Personalizzazioni per AWS Control Tower (AWS Solutions Library)
- Configurazione di un AWS ambiente multi-account sicuro e scalabile (Prescriptive Guidance)AWS

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Migliora le prestazioni operative abilitando Amazon DevOps Guru su più regioni AWS, account e OUs con AWS CDK

Dott. Rahul Sharad Gaikwad, Amazon Web Services

Riepilogo

Questo modello illustra i passaggi per abilitare il servizio Amazon DevOps Guru su più regioni, account e unità organizzative di Amazon Web Services (AWS) (OUs) utilizzando l'AWS Cloud Development Kit (AWS CDK) in. TypeScript Puoi utilizzare AWS CDK stacks per distribuire AWS dall'account CloudFormation StackSets AWS amministratore (primario) per abilitare DevOps Amazon Guru su più account, invece di accedere a ciascun account e DevOps abilitare Guru singolarmente per ogni account.

Amazon DevOps Guru offre funzionalità operative di intelligenza artificiale (AIOps) per aiutarti a migliorare la disponibilità delle tue applicazioni e risolvere più rapidamente i problemi operativi. DevOps Guru riduce il lavoro manuale applicando consigli basati sull'apprendimento automatico (ML), senza richiedere alcuna esperienza di machine learning. DevOps Guru analizza le risorse e i dati operativi. Se rileva anomalie, fornisce metriche, eventi e consigli per aiutarti a risolvere il problema.

Questo modello descrive tre opzioni di distribuzione per abilitare Amazon DevOps Guru:

- Per tutte le risorse, raggruppa le risorse su più account e regioni
- Per tutte le risorse dello stack OUs
- Per risorse stack specifiche su più account e regioni

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- AWS Command Line Interface (AWS CLI), installata e configurata. (Vedi Installazione, aggiornamento e disinstallazione dell'AWS CLI nella documentazione dell'interfaccia a riga di comando di AWS.)
- AWS CDK Toolkit, installato e configurato. (Vedi AWS CDK Toolkit nella documentazione di AWS CDK.)

- Node Package Manager (npm), installato e configurato per AWS CDK in. TypeScript (Vedi Download e installazione di Node.js e npm nella documentazione di npm.)
- Python3 installato e configurato, per eseguire uno script Python per iniettare traffico nell'applicazione serverless di esempio. (Vedi Configurazione e utilizzo di Python nella documentazione di Python.)
- Pip, installato e configurato per installare la libreria di richieste Python. (Vedi le istruzioni di installazione di pip sul sito web.) PyPI

Versioni del prodotto

- AWS CDK Toolkit versione 1.107.0 o successiva
- npm versione 7.9.0 o successiva
- Node.js versione 15.3.0 o successiva

Architecture

Tecnologie

L'architettura di questo modello include i seguenti servizi:

- Amazon DevOps Guru
- AWS CloudFormation
- Gateway Amazon API
- AWS Lambda
- Amazon DynamoDB
- Amazon CloudWatch
- AWS CloudTrail

Stack CDK AWS

Il modello utilizza i seguenti stack AWS CDK:

- `CdkStackSetAdminRole`— Crea un ruolo di amministratore di AWS Identity and Access management (IAM) per stabilire una relazione di fiducia tra l'amministratore e gli account di destinazione.

- `CdkStackSetExecRole`— Crea un ruolo IAM per fidarsi dell'account amministratore.
- `CdkDevopsGuruStackMultiAccReg`— Abilita DevOps Guru su più regioni AWS e account per tutti gli stack e configura le notifiche Amazon Simple Notification Service (Amazon SNS).
- `CdkDevopsGuruStackMultiAccRegSpecStacks`— Abilita DevOps Guru su più regioni AWS e account per stack specifici e configura le notifiche Amazon SNS.
- `CdkDevopsguruStackOrgUnit`— Abilita DevOps Guru e OUs configura le notifiche Amazon SNS.
- `CdkInfrastructureStack`— Implementa componenti applicativi serverless di esempio come API Gateway, Lambda e DynamoDB nell'account amministratore per dimostrare l'iniezione di errori e la generazione di approfondimenti.

Architettura applicativa di esempio

Il diagramma seguente illustra l'architettura di un'applicazione serverless di esempio che è stata distribuita su più account e regioni. Il modello utilizza l'account amministratore per distribuire tutti gli stack CDK AWS. Inoltre, utilizza l'account amministratore come uno degli account di destinazione per la configurazione di Guru. DevOps

1. Quando DevOps Guru è abilitato, prima definisce il comportamento di ogni risorsa e poi acquisisce i dati operativi dalle metriche fornite. CloudWatch
2. Se rileva un'anomalia, la mette in correlazione con gli eventi generati e genera un'analisi approfondita. CloudTrail
3. L'analisi fornisce una sequenza di eventi correlata insieme a raccomandazioni prescritte per consentire all'operatore di identificare la risorsa responsabile.
4. Amazon SNS invia messaggi di notifica all'operatore.

Automazione e scalabilità

Il GitHub repository fornito con questo modello utilizza AWS CDK come strumento Infrastructure as Code (IaC) per creare la configurazione per questa architettura. AWS CDK ti aiuta a orchestrare le risorse e abilitare DevOps Guru su più account AWS, regioni e. OUs

Tools (Strumenti)

Servizi AWS

- AWS CDK — AWS Cloud Development Kit (AWS CDK) ti aiuta a definire la tua infrastruttura cloud come codice in uno dei cinque linguaggi di programmazione supportati: TypeScript, JavaScript Python, Java e C#.
- AWS CLI — AWS Command Line Interface (AWS CLI) è uno strumento unificato che fornisce un'interfaccia a riga di comando coerente per interagire con i servizi e le risorse AWS.

Codice

Il codice sorgente di questo pattern è disponibile nel GitHub repository Amazon DevOps Guru CDK Samples. Il codice CDK AWS è scritto in TypeScript. Per clonare e utilizzare il repository, segui le istruzioni nella sezione successiva.

> **Important**
>
> Alcune delle storie di questo modello includono esempi di comandi AWS CDK e AWS CLI formattati per Unix, Linux e macOS. Per Windows, sostituisci il carattere di continuazione della barra rovesciata (\) alla fine di ogni riga con un accento circonflesso (^).

Epiche

Prepara le risorse AWS per la distribuzione

Operazione	Description	Competenze richieste

Configura i profili denominati AWS.	Configura i tuoi profili denominati AWS come segue per distribuire gli stack in un ambiente con più account.
	
	Per l'account amministratore:
	
	`$aws configure --profile administrator`
	`AWS Access Key ID [****]: <your-administrator-access-key-ID>`
		DevOps ingegnere

Operazione	Description	Competenze richieste

	`AWS Secret Access Key [****]: <your-administrator-secret-access-key>`
	`Default region name [None]: <your-administrator-region>`
	`Default output format [None]: json`
	
	Per l'account di destinazione:
	
	`$aws configure --profile target`
	`AWS Access Key ID [****: <your-target-access-key-ID>`
	`AWS Secret Access Key [****]: <your-target-secret-access-key>`
	`Default region name [None]: <your-target-region>`
	`Default output format [None]: json`
	
	Per ulteriori informazioni, consulta Using named profiles nella documentazione AWS CLI.	

Operazione	Description	Competenze richieste
Verifica le configurazioni dei profili AWS.	(Facoltativo) Puoi verificare le configurazioni del tuo profilo AWS nei `config` file `credentials` and seguendo le istruzioni in Impostare e visualizzare le impostazioni di configurazione nella documentazione dell'interfaccia a riga di comando di AWS.	DevOps ingegnere
Verifica la versione di AWS CDK.	Verifica la versione di AWS CDK Toolkit eseguendo il seguente comando: `$cdk --version` Questo modello richiede la versione 1.107.0 o successiva. Se disponi di una versione precedente di AWS CDK, segui le istruzioni nella documentazione di AWS CDK per aggiornarla.	DevOps ingegnere
Clona il codice del progetto.	Clona il GitHub repository per questo pattern usando il comando: `$git clone https://github.com/aws-samples/amazon-devopsguru-cdk-samples.git`	DevOps ingegnere

Operazione	Description	Competenze richieste

Installa le dipendenze dei pacchetti e compila i TypeScript file.	Installa le dipendenze del pacchetto e compila TypeScript i file eseguendo i seguenti comandi:
	`$cd amazon-devopsguru-cdk-samples`
	`$npm install`
	`$npm fund`
	Questi comandi installano tutti i pacchetti dal repository di esempio.
	Important
	Se ricevi errori sui pacchetti mancanti, usa uno dei seguenti comandi:
	`$npm ci`
	oppure
	`$npm install -g @aws-cdk/<package-name>`
	Puoi trovare l'elenco dei nomi e delle versioni dei pacchetti nella `Dependencies` sezione del `/amazon-devopsguru-cdk-samples/package.json` file.
		DevOps ingegnere

Operazione	Description	Competenze richieste
	Per ulteriori informazioni, vedere npm ci e npm install nella documentazione di npm.	

Crea (sintetizza) gli stack CDK AWS

Operazione	Description	Competenze richieste
Configura un indirizzo e-mail per le notifiche di Amazon SNS.	Segui questi passaggi per fornire un indirizzo e-mail per le notifiche di Amazon SNS: 1. Modifica i file `/amazon-devopsguru-cdk-samples/lib/cdk-devopsguru-multi-acc-reg-stack.ts` e. `/amazon-devopsguru-cdk-samples/lib/cdk-devopsguru-org-uni-stack.ts` 2. Nella `Subscription` sezione`DevOpsGuruTopic`, aggiorna il `Endpoint` parametro con il tuo indirizzo email. 3. Salva e chiudi i file.	DevOps ingegnere
Costruisci il codice del progetto.	Crea il codice del progetto e sintetizza gli stack eseguendo il comando: `npm run build && cdk synth`	DevOps ingegnere

Operazione	Description	Competenze richieste

	Verrà visualizzato un output simile al seguente:
	
	`$npm run build && cdk synth`
	`> cdk-devopsguru@0.1.0 build`
	`> tsc`
	`Successfully synthesized to ~/amazon-devopsguru-cdk-samples/cdk.out`
	`Supply a stack id (CdkDevopsGuruStackMultiAccReg,CdkDevopsGuruStackMultiAccRegSpecStacks, CdkDevopsguruStackOrgUnit, CdkInfrastructureStack, CdkStackSetAdminRole, CdkStackSetExecRole) to display its template.`
	
	Per ulteriori informazioni e passaggi, consulta La tua prima app AWS CDK nella documentazione di AWS CDK.	

Operazione	Description	Competenze richieste

Elenca gli stack CDK AWS.	Esegui il comando seguente per elencare tutti gli stack CDK AWS:
	`$cdk list`
	Il comando visualizza il seguente elenco:
	`CdkDevopsGuruStack`
	`MultiAccReg`
	`CdkDevopsGuruSt`
	`ackMultiAccRegSpec`
	`Stacks`
	`CdkDevopsguruStackOr`
	`gUnit`
	`CdkInfrastructureStac`
	`k`
	`CdkStackSetAdminRole`
	`CdkStackSetExecRole`
		DevOps ingegnere

Opzione 1: abilita DevOps Guru per tutte le risorse dello stack su più account

Operazione	Description	Competenze richieste
Implementa gli stack CDK AWS per creare ruoli IAM.	Questo modello utilizza AWS CloudFormation StackSets per eseguire operazioni di stack su più account. Se stai creando il tuo primo set di stack, devi creare i seguenti ruoli IAM per ottenere le autorizzazioni richieste nei tuoi account AWS:	DevOps ingegnere

Operazione	Description	Competenze richieste
	• `AWSCloudFormationStackSetAdministrationRole` • `AWSCloudFormationStackSetExecutionRole` **Note** I ruoli devono avere questi nomi esatti. 1. Crea il `AWSCloudFormationStackSetAdministrationRole` ruolo IAM nell'account amministratore (primario) eseguendo il seguente comando CLI: `$cdk deploy CdkStackSetAdminRole --profile administrator` 2. Crea il `AWSCloudFormationStackSetExecutionRole` ruolo IAM in tutti gli account di destinazione in cui desideri eseguire le istanze dello stack. Per creare questo ruolo, esegui questi comandi CLI:	

Operazione	Description	Competenze richieste

	`$cdk deploy CdkStackSetExecRole \ --parameters AdministratorAccountId=<administrator-account-ID> \ --profile administrator`
	
	`$cdk deploy CdkStackSetExecRole \ --parameters AdministratorAccountId=<administrator-account-ID> \ --profile target`
	
	Per ulteriori informazioni, consulta Concedere autorizzazioni autogestite nella documentazione CloudFormation AWS.	

Operazione	Description	Competenze richieste
Implementa lo stack AWS CDK per abilitare DevOps Guru su più account.	Lo `CdkDevopsGuruStack MultiAccReg` stack CDK di AWS crea set di stack per distribuire istanze di stack su più account e regioni. Per distribuire lo stack, esegui il seguente comando CLI con i parametri specificati: `$cdk deploy CdkDevops GuruStackMultiAccReg \ --profile administr ator \ --parameters AdministratorAccou ntId=<administrator- account-ID> \ --parameters TargetAccountId=<t arget-account-ID> \ --parameters RegionIds="<region -1>,<region-2>"` Attualmente Amazon DevOps Guru è disponibile nelle regioni AWS elencate nelle domande frequenti su DevOps Guru.	DevOps ingegnere

Opzione 2: abilita DevOps Guru per tutte le risorse dello stack OUs

Operazione	Description	Competenze richieste
Estrarre OU IDs.	Sulla console AWS Organizations, identifica IDs le unità organizzative in cui desideri abilitare DevOps Guru.	DevOps ingegnere
Abilita le autorizzazioni gestite dal servizio per. OUs	Se utilizzi AWS Organizations per la gestione degli account, devi concedere le autorizzazioni gestite dal servizio per abilitare Guru. DevOps Invece di creare i ruoli IAM manualmente, utilizza ruoli di accesso affidabile e collegati ai servizi basati sull'organizzazione (). SLRs	DevOps ingegnere
Implementa lo stack AWS CDK per abilitare DevOps Guru Across. OUs	Lo `CdkDevopsguruStack OrgUnit` stack CDK AWS abilita il servizio DevOps Guru in tutto il mondo. OUs Per distribuire lo stack, esegui il seguente comando con i parametri specificati: `$cdk deploy CdkDevops guruStackOrgUnit \ --profile administr ator \ --parameters RegionIds="<region -1>,<region-2>" \ --parameters OrganizationalUnit Ids="<OU-1>,<OU-2>"`	DevOps ingegnere

Opzione 3: abilita DevOps Guru per risorse stack specifiche su più account

Operazione	Description	Competenze richieste

Implementa gli stack CDK AWS per creare ruoli IAM.	Se non hai ancora creato i ruoli IAM richiesti mostrati nella prima opzione, fallo prima:
	
	1. Crea il `AWSCloudFormationStackSetAdministrationRole` ruolo IAM nell'account amministratore (primario) eseguendo il seguente comando CLI:
	
	`$cdk deploy CdkStackSetAdminRole --profile administrator`
	
	2. Crea il `AWSCloudFormationStackSetExecutionRole` ruolo IAM in tutti gli account di destinazione in cui desideri eseguire le istanze dello stack. Per creare questo ruolo, esegui i comandi CLI:
	
	`$cdk deploy CdkStackSetExecRole \ --parameters AdministratorAccountId=<administrator-account-ID> \ --profile administrator`
		DevOps ingegnere

Operazione	Description	Competenze richieste
	`$cdk deploy CdkStackSetExecRole \ --parameters AdministratorAccountId=<administrator-account-ID> \ --profile target` Per ulteriori informazioni, consulta Concedere autorizzazioni autogestite nella documentazione CloudFormation AWS.	
Elimina le pile esistenti.	Se hai già utilizzato la prima opzione per abilitare DevOps Guru per tutte le risorse dello stack, puoi eliminare il vecchio stack usando il seguente comando: `$cdk destroy CdkDevopsGuruStackMultiAccReg --profile administrator` In alternativa, puoi modificare il *`RegionIds`* parametro quando ridistribuisci lo stack per evitare che si verifichi un errore Stacks already exist.	DevOps ingegnere

Operazione	Description	Competenze richieste
Aggiorna lo stack CDK AWS con un elenco di stack.	1. Modificare il file `/amazon-devopsguru-cdk-samples/lib/cdk-devopsguru-multi-acc-reg-spec-stack.ts` . 2. Sotto `Resources` ,, `CloudFormation StackNames` , elenca gli stack per i quali desideri abilitare Guru. DevOps A scopo dimostrativo, il parametro specifica lo `CdkInfrastructureStack` stack, ma puoi modificare questa voce in base alle tue esigenze. 3. Salva e chiudi il file. 4. Per sintetizzare e aggiornare il modello dello stack, esegui: `$cdk synth`	Ingegnere dei dati

Operazione	Description	Competenze richieste
Implementa lo stack AWS CDK per abilitare DevOps Guru per risorse stack specifiche su più account.	Lo `CdkDevopsGuruStack MultiAccRegSpecSta cks` stack AWS CDK consente a DevOps Guru di utilizzare risorse stack specifiche su più account. Per distribuire lo stack, esegui il seguente comando: `$cdk deploy CdkDevops GuruStackMultiAccR egSpecStacks \ --profile administr ator \ --parameters AdministratorAccou ntId=<administrator- account-ID> \ --parameters TargetAccountId=<t arget-account-ID> \ --parameters RegionIds="<region -1>,<region-2>"` **Note** Se in precedenza hai distribuito questo stack per l'opzione 1, modifica il `RegionIds` parametro (assicurandoti di scegliere tra le regioni disponibili) per evitare che si	DevOps ingegnere

Operazione	Description	Competenze richieste
	verifichi un errore Stacks already exist.	

Implementa lo stack di infrastruttura AWS CDK

Operazione	Description	Competenze richieste
Implementa lo stack di infrastruttura serverless di esempio.	Lo `CdkInfrastructureStack` stack CDK di AWS implementa componenti serverless come API Gateway, Lambda e una tabella DynamoDB per illustrare le intuizioni di Guru. DevOps Per distribuire lo stack, esegui il seguente comando: `$cdk deploy CdkInfrastructureStack --profile administrator`	DevOps ingegnere
Inserisci record di esempio in DynamoDB.	Esegui il comando seguente per popolare la tabella DynamoDB con record di esempio. Fornisci il percorso corretto per lo script. `populate-shops-dynamodb-table.json` `$aws dynamodb batch-write-item \ --request-items file://scripts/populate-shops-dynamodb-table.json \`	DevOps ingegnere

Operazione	Description	Competenze richieste

	`--profile administrator`
	
	Il comando visualizza il seguente output:
	
	`{ "UnprocessedItems": {} }`	

Operazione	Description	Competenze richieste

Verifica i record inseriti in DynamoDB.	Per verificare che la tabella DynamoDB includa i record di esempio `populate-shops-dynamodb-table.json` del file, accedi all'URL `ListRestApiEndpointMonitorOperator` dell'API, che viene pubblicato come output dello stack CDK AWS. Puoi trovare questo URL anche nella scheda Outputs della CloudFormation console AWS per lo `CdkInfrastructureStack` stack. L'output di AWS CDK sarebbe simile al seguente:
	
	`CdkInfrastructureStack.CreateRestApiMonitorOperatorEndpointD1D00045 = https://oure17c5vob.execute-api.<your-region>.amazonaws.com/prod/`
	
	`CdkInfrastructureStack.ListRestApiMonitorOperatorEndpointABBDB8D8 = https://cdff8icfrn4.execute-api.<your-region>.amazonaws.com/prod/`
		DevOps ingegnere

Operazione	Description	Competenze richieste
Attendi che le risorse completino la linea di base.	Questo stack serverless ha poche risorse. Ti consigliamo di attendere 2 ore prima di eseguire i passaggi successiv i. Se hai distribuito questo stack in un ambiente di produzione, potrebbero essere necessarie fino a 24 ore per completare la baseline, a seconda del numero di risorse che hai selezionato per il monitoraggio in Guru. DevOps	DevOps ingegnere

Genera DevOps approfondimenti su Guru

Operazione	Description	Competenze richieste
Aggiorna lo stack di infrastru ttura AWS CDK.	Per provare DevOps Guru Insights, puoi apportare alcune modifiche alla configura zione per riprodurre un tipico problema operativo. 1. Modificare il file `/amazon-devopsguru-cdk-samples/lib/infrastructure-stack.ts` . 2. Nella `DDB Table` sezione, modifica la capacità di lettura per la tabella DynamoDB da 5 a 1. 3. Salva e chiudi il file.	DevOps ingegnere

Operazione	Description	Competenze richieste

	4. Esegui i seguenti comandi per sintetizzare e distribuire lo stack di infrastruttura AWS CDK aggiornato:
	`$cdk synth`
	`$cdk deploy CdkInfrastructureStack --profile administrator`	

Operazione	Description	Competenze richieste
Inietta richieste HTTP sull'API.	Inietta il traffico in ingresso sotto forma di richieste HTTP sull'API: `ListRestApiMonitorOperatorEndpointxxxx` 1. Modifica lo script Python. `/amazon-devopsguru-cdk-samples/scripts/sendAPIRequest.py` 2. Aggiorna la `url` variabile con il link API per`ListRestApiMonitorOperatorEndpointxxxx` . Puoi trovare questo URL nell'output del comando AWS CDK deploy o nella console AWS Cloudformation, nella scheda Outputs dello stack. 3. Salva e chiudi il file. 4. Esegui lo script Python usando il comando: `$python sendAPIRequest.py` 5. Assicurati di ottenere un codice di stato 200. 6. Potrebbe essere necessario eseguire lo script su più terminali (preferibilmente quattro) per immettere	DevOps ingegnere

Operazione	Description	Competenze richieste
	traffico a una velocità elevata. 7. Dopo che lo script è stato eseguito per circa 10 minuti in un ciclo, puoi visualizzare una panoramica operativa sulla console DevOps Guru.	
Rivedi le informazioni su DevOps Guru.	In condizioni standard, la dashboard di DevOps Guru mostra zero nel contatore delle informazioni in corso. Se rileva un'anomalia, genera un avviso sotto forma di intuizione. Nel riquadro di navigazione, scegli Insights per visualizzare i dettagli dell'anomalia, tra cui una panoramica, metriche aggregate, eventi pertinenti e consigli. Per ulteriori informazioni sulla revisione degli approfondimenti, consulta il post del blog Gaining operating insights with AIOps Amazon DevOps Guru.	DevOps ingegnere

Eliminazione

Operazione	Description	Competenze richieste
Pulisci ed elimina le risorse.	Dopo aver seguito questo schema, dovresti rimuovere le risorse che hai creato per evitare di incorrere in	DevOps ingegnere

Operazione	Description	Competenze richieste

	ulteriori addebiti. Esegui questi comandi:
	
	`$cdk destroy CdkDevopsGuruStackMultiAccReg --profile administrator`
	`$cdk destroy CdkDevopsguruStackOrgUnit --profile administrator`
	`$cdk destroy CdkDevopsGuruStackMultiAccRegSpecStacks --profile administrator`
	`$cdk destroy CdkInfrastructureStack --profile administrator`
	`$cdk destroy CdkStackSetAdminRole --profile administrator`
	`$cdk destroy CdkStackSetExecRole --profile administrator`
	`$cdk destroy CdkStackSetExecRole --profile target`	

Risorse correlate

- Ottenere informazioni operative AIOps con Amazon DevOps Guru
- Configura facilmente Amazon DevOps Guru su più account e regioni utilizzando AWS CloudFormation StackSets
- DevOps Workshop per guru

Gestisci i set di autorizzazioni per più account utilizzando Account Factory for Terraform

Anand Krishna Varanasi e Siamak Heshmati, Amazon Web Services

Riepilogo

Questo modello ti aiuta a integrare AWS Control Tower Account Factory Terraform (AFT) con AWS IAM Identity Centerper configurare autorizzazioni multiple Account AWS su larga scala. Questo approccio utilizza AWS Lambda funzioni personalizzate per automatizzare l'assegnazione dei set di autorizzazioni ai Account AWS quali vengono gestiti come organizzazione. Questo semplifica il processo perché non richiede l'intervento manuale del team di progettazione della piattaforma. Questa soluzione può migliorare l'efficienza operativa, la sicurezza e la coerenza. Promuove un processo di onboarding sicuro e standardizzato per AWS Control Tower, rendendolo indispensabile per le aziende che danno priorità all'agilità e all'affidabilità della propria infrastruttura cloud.

Prerequisiti e limitazioni

Prerequisiti

- Account AWS, gestito tramite. AWS Control Tower Per ulteriori informazioni, vedi Guida introduttiva AWS Control Tower.
- Account Factory for Terraform, distribuito in un account dedicato nel tuo ambiente. Per ulteriori informazioni, consulta Deploy AWS Control Tower Account Factory for Terraform.
- Un'istanza IAM Identity Center, configurata nel tuo ambiente. Per ulteriori informazioni, consulta Guida introduttiva a IAM Identity Center.
- Un gruppo IAM Identity Center attivo, configurato. Per ulteriori informazioni, consulta Aggiungere gruppi al tuo IAM Identity Center directory.
- Python versione 3.9 o successiva, installata

Limitazioni

- Questa soluzione può essere utilizzata solo con account gestiti tramite. AWS Control Tower Questa soluzione viene implementata utilizzando Account Factory for Terraform.
- Questo modello non include istruzioni per configurare la federazione delle identità con una fonte di identità. Per ulteriori informazioni su come completare questa configurazione, consulta i tutorial sulle fonti di identità di IAM Identity Center nella documentazione di IAM Identity Center.

Architecture

Panoramica di AFT

AFT imposta una pipeline Terraform che ti aiuta a fornire e personalizzare i tuoi account. AWS Control Tower AFT segue un GitOps modello che automatizza i processi di fornitura degli account in. AWS Control Tower Si crea un file Terraform per la richiesta di account e lo si archivia nel repository. Questo avvia il flusso di lavoro AFT per il provisioning degli account. Una volta completato il provisioning dell'account, AFT può eseguire automaticamente passaggi di personalizzazione aggiuntivi. Per ulteriori informazioni, consultate l'architettura AFT nella AWS Control Tower documentazione.

AFT fornisce i seguenti repository principali:

- `aft-account-request`— Questo repository contiene codice Terraform da creare o aggiornare. Account AWS
- `aft-account-customizations`- Questo repository contiene codice Terraform per creare o personalizzare risorse per account.
- `aft-global-customizations`— Questo repository contiene codice Terraform per creare o personalizzare risorse per tutti gli account, su larga scala.
- `aft-account-provisioning-customizations`- Questo repository gestisce le personalizzazioni che vengono applicate solo a account specifici creati e gestiti con AFT. Ad esempio, è possibile utilizzare questo repository per personalizzare le assegnazioni di utenti o gruppi in IAM Identity Center o per automatizzare la chiusura degli account.

Panoramica della soluzione

Questa soluzione personalizzata include una macchina a AWS Step Functions stati e una AWS Lambda funzione che assegna set di autorizzazioni a utenti e gruppi per più account. La macchina a stati distribuita tramite questo modello funziona insieme alla macchina a stati AFT preesistente. `aft_account_provisioning_customizations` Un utente invia una richiesta per aggiornare le assegnazioni di utenti e gruppi di IAM Identity Center quando ne Account AWS viene creata una nuova o dopo la creazione dell'account. Lo fanno inserendo una modifica nel repository. `aft-account-request` La richiesta di creazione o aggiornamento di un account avvia uno stream in Amazon DynamoDB Streams. Questo avvia la funzione Lambda, che aggiorna gli utenti e i gruppi di IAM Identity Center per la destinazione. Account AWS

Di seguito è riportato un esempio dei parametri che è possibile fornire nella funzione Lambda per l'assegnazione di set di autorizzazioni a utenti e gruppi target:

```
custom_fields = {
    "InstanceArn"         = "<Organization ID>",
    "PermissionSetArn"    = "<Permission set ARN>",
    "PrincipalId"         = "<Principal ID>",
  }
```

Di seguito sono riportati i parametri di questa dichiarazione:

- `InstanceArn`— L'Amazon Resource Name (ARN) dell'organizzazione
- `PermissionSetArn`— L'ARN del set di autorizzazioni
- `PrincipalId`— L'identificatore di un utente o di un gruppo in IAM Identity Center a cui verrà applicato il set di autorizzazioni

> **Note**
>
> È necessario creare il set di autorizzazioni di destinazione, gli utenti e i gruppi prima di eseguire questa soluzione.

Sebbene il `InstanceArn` valore debba rimanere coerente, è possibile modificare la funzione Lambda per assegnare più set di autorizzazioni a più identità di destinazione. I parametri per i set di autorizzazioni devono terminare con e i parametri per utenti e gruppi devono terminare con. `PermissionSetArn` `PrincipalId` È necessario definire entrambi gli attributi. Di seguito è riportato un esempio di come definire più set di autorizzazioni e indirizzare utenti e gruppi:

```
custom_fields = {
    "InstanceArn"                    = "<Organization ID>",
    "AdminAccessPermissionSetArn"    = "<Admin privileges permission set ARN>",
    "AdminAccessPrincipalId"         = "<Admin principal ID>",
    "ReadOnlyAccessPermissionSetArn" = "<Read-only privileges permission set ARN>",
    "ReadOnlyAccessPrincipalId"      = "<Read-only principal ID>",
  }
```

Il diagramma seguente mostra un step-by-step flusso di lavoro che illustra come la soluzione aggiorna i set di autorizzazioni per utenti e gruppi nella destinazione Account AWS su larga scala.

Quando l'utente avvia una richiesta di creazione di account, AFT avvia la macchina a stati `aft-account-provisioning-framework` Step Functions. Questa macchina a stati avvia la funzione `extract-alternate-sso` Lambda. La funzione Lambda assegna set di autorizzazioni a utenti e gruppi nella destinazione. Account AWS Questi utenti o gruppi possono provenire da qualsiasi fonte di identità configurata in IAM Identity Center. Esempi di fonti di identità includono Okta, Active Directory o Ping Identity.

Il diagramma mostra il seguente flusso di lavoro quando vengono creati nuovi account:

1. Un utente inserisce una `custom_fields` modifica nel `aft-account-request` repository.
2. AWS CodePipeline avvia un AWS CodeBuild processo che registra i metadati definiti dall'utente nella tabella Amazon `aft-request-audit` DynamoDB. Questa tabella contiene attributi per registrare i metadati definiti dall'utente. L'`ddb_event_name`attributo definisce il tipo di operazione AFT:
   - Se il valore è`INSERT`, la soluzione assegna le autorizzazioni impostate alle identità di destinazione al momento della creazione della nuova Account AWS .
   - Se il valore è`UPDATE`, la soluzione assegna le autorizzazioni impostate alle identità di destinazione dopo la creazione di. Account AWS
3. Amazon DynamoDB Streams avvia `aft_alternate_sso_extract` la funzione Lambda.
4. La funzione `aft_alternate_sso_extract` Lambda assume un ruolo AWS Identity and Access Management (IAM) nell'account di gestione. AWS Control Tower
5. La funzione Lambda assegna i set di autorizzazioni agli utenti e ai gruppi di destinazione effettuando una chiamata API AWS SDK per Python (Boto3) create_account_assignment a IAM Identity Center. Recupera il set di autorizzazioni e le assegnazioni di identità dalla tabella Amazon `aft-request-audit` DynamoDB.
6. Al termine del flusso di lavoro Step Functions, i set di autorizzazioni vengono assegnati alle identità di destinazione.

Automazione e scalabilità

AFT opera su larga scala utilizzando CodePipeline, Servizi AWS ad esempio AWS CodeBuild, DynamoDB e Lambda, che sono altamente scalabili. Per un'ulteriore automazione, puoi integrare questa soluzione con un sistema di gestione dei ticket o dei problemi, come Jira. Per ulteriori informazioni, consulta la sezione Informazioni aggiuntive di questo modello.

Tools (Strumenti)

Servizi AWS

- Account Factory for Terraform (AFT) è lo strumento principale di questa soluzione. Il `aft-account-provisioning-customizations` repository contiene il codice Terraform per la creazione di personalizzazioni per Account AWS, ad esempio assegnazioni personalizzate di utenti o gruppi di IAM Identity Center.
- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS Step Functionsè un servizio di orchestrazione senza server che consente di combinare AWS Lambda funzioni e altro Servizi AWS per creare applicazioni aziendali critiche.

Altri strumenti

- Python è un linguaggio di programmazione per computer generico.
- Terraform è uno strumento Infrastructure as Code (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.

Archivio di codici

Il repository di codice per AFT è disponibile nel repository GitHub AWS Control Tower Account Factory for Terraform. Il codice per questo pattern è disponibile nel Govern SSO Assignments per l'utilizzo del Account AWS repository Account Factory for Terraform (AFT).

Best practice

- Comprendi il modello di responsabilità condivisa.AWS
- Segui le raccomandazioni di sicurezza per AWS Control Tower. Per ulteriori informazioni, consulta Sicurezza in AWS Control Tower.
- Segui il principio del privilegio minimo. Per ulteriori informazioni, consulta Applicazione delle autorizzazioni del privilegio minimo.
- Crea set di autorizzazioni e ruoli IAM specifici e mirati per gruppi e unità aziendali.

Epiche

Implementazione della soluzione

Operazione	Description	Competenze richieste

Crea un ruolo IAM.	Nell'account di AWS Control Tower gestione, usa Terraform per creare un ruolo IAM. Questo ruolo ha un accesso tra account e una politica di fiducia che consente l'accesso federato dal provider di identità. Dispone inoltre delle autorizzazioni per concedere l'accesso ad altri account tramite. AWS Control Tower La funzione Lambda assumerà questo ruolo. Esegui questa operazione:
	1. Scarica il `AFTCrossAccountRole.tf` file dal repository del GitHub codice.
	2. Modificate il `AFTCrossAccountRole.tf` file in base alle esigenze del vostro AWS ambiente.
	3. In Terraform, inserisci i seguenti comandi per creare questo ruolo IAM:
	`terraform init`
	`terraform plan`
	`terraform apply`
		AWS DevOps, architetto del cloud

Operazione	Description	Competenze richieste
	4. Verifica che il ruolo sia stato distribuito correttamente e abbia l'accesso tra account diversi previsto.	

Operazione	Description	Competenze richieste
Personalizza la soluzione per il tuo ambiente.	1. Immettere il seguente comando per clonare il Govern SSO Assignments per l'utilizzo del Account AWS repository Account Factory for Terraform (AFT) sulla workstation locale. `git clone https://github.com/aws-samples/aft-custom-sso-assignment.git` 2. Nella cartella, apri il file. `aft-account-provisioning-customizations/terraform variables.tf` 3. Modificate le variabili in base alle esigenze del vostro ambiente. 4. Salvare e chiudere il file `variables.tf` . 5. Apri il `account-request.tf` file nel tuo `aft-account-request` repository. 6. Modifica i `custom_fields` parametri per definire i set di autorizzazioni e indirizzare utenti e gruppi. Per ulteriori informazioni, vedere la sezione Architettura di questo modello.	AWS DevOps, architetto del cloud

Operazione	Description	Competenze richieste
	7. Salvare e chiudere il file `account-request.tf` .	
Distribuire la soluzione.	1. Nel repository clonato, copia il contenuto della `terraform` cartella e incollalo nella `terraform` cartella del repository. `aft-account-provisioning-customizations` 2. Nell'account di gestione AFT, avvia la pipeline. `ct-aft-account-provisioning-customizations` Questo implementa la soluzione personalizzata. Per istruzioni, consulta Avviare una pipeline in. CodePipeline 3. Verifica che le risorse siano state distribuite correttamente nell'account di gestione AFT.	AWS DevOps, architetto del cloud

Operazione	Description	Competenze richieste
Configura una connessione al repository di codice.	Configura una connessione tra il repository di codice in cui archivierai i file di configura zione e il tuo. Account AWS Per istruzioni, consulta la sezione Aggiungere fornitori di sorgenti di terze parti alle pipeline utilizzate CodeConne ctions nella AWS CodePipel ine documentazione.	AWS DevOps, architetto del cloud

Usa la soluzione

Operazione	Description	Competenze richieste
Avvia la pipeline AFT per distribuire un nuovo account.	Segui le istruzioni riportate in Fornire un nuovo account con AFT per avviare la pipeline che ne crea uno nuovo Account AWS nel tuo ambiente. AWS Control Tower Attendi il completamento del processo di creazione dell'account.	AWS DevOps, architetto del cloud
Convalida le modifiche.	1. Apri la AWS IAM Identity Center console. 2. Nell'elenco degli account, scegli l'account appena creato. 3. Verifica che i set di autorizzazioni siano stati assegnati per concedere	AWS DevOps, architetto del cloud

Operazione	Description	Competenze richieste
	l'accesso agli utenti e ai gruppi target.	

risoluzione dei problemi

Problema	Soluzione
L'assegnazione del set di autorizzazioni non funziona.	Assicurati che l'ARN del gruppo, l'id dell'orga nizzazione e i parametri Lambda siano corretti. Per esempi, consulta la sezione Panoramica della soluzione di questo modello.
L'aggiornamento del codice nel repository non avvia la pipeline.	Questo problema è correlato alla connettiv ità tra l'utente Account AWS e il repository. Nel Console di gestione AWS, verifica che la connessione sia attiva. Per ulteriori informazi oni, consulta GitHub le connessioni nella AWS CodePipeline documentazione.

Informazioni aggiuntive

Integrazione con uno strumento di gestione dei ticket

Puoi scegliere di integrare questa soluzione con uno strumento di gestione dei ticket o dei problemi, come Jira o. ServiceNow Il diagramma seguente mostra un esempio di flusso di lavoro per questa opzione. È possibile integrare lo strumento di gestione dei ticket con gli archivi delle soluzioni AFT utilizzando i connettori dello strumento. Per i connettori Jira, consulta Integrare Jira con. GitHub Per i ServiceNow connettori, vedi Integrazione con. GitHub Puoi persino creare soluzioni personalizzate che richiedono agli utenti di fornire un ID del ticket come parte dell'approvazione della pull request. Se viene approvata una richiesta per crearne una nuova Account AWS utilizzando AFT, quell'evento potrebbe avviare un flusso di lavoro che aggiunge campi personalizzati al `aft-account-request` GitHub repository. Puoi progettare qualsiasi flusso di lavoro personalizzato che soddisfi i requisiti del tuo caso d'uso.

Il diagramma mostra il flusso di lavoro seguente:

1. Gli utenti richiedono l'assegnazione di un set di autorizzazioni personalizzato in uno strumento di gestione dei ticket, come Jira.
2. Dopo l'approvazione del caso, inizia un flusso di lavoro per aggiornare l'assegnazione del set di autorizzazioni. (Facoltativo) È possibile utilizzare i plugin per l'automazione personalizzata di questo passaggio.
3. Gli operatori inviano il codice Terraform con i parametri aggiornati del set di autorizzazioni al `aft-account-request` repository in un ramo di sviluppo o di funzionalità.
4. GitHub Le azioni vengono AWS CodeBuild avviate utilizzando una chiamata OpenID Connect (OIDC). CodeBuild esegue scansioni di sicurezza dell'infrastruttura come codice (IaC) utilizzando strumenti come tfsec e checkov. Avvisa gli operatori di eventuali violazioni della sicurezza.
5. Se non viene rilevata alcuna violazione, GitHub Actions crea una pull request automatica e assegna una revisione del codice ai proprietari del codice. Crea anche un tag per la pull request.
6. Se il proprietario del codice approva la revisione del codice, viene avviato un altro flusso di lavoro GitHub Actions. Controlla gli standard delle pull request, tra cui:
   - Se il titolo della pull request soddisfa i requisiti.
   - Se il corpo della pull request contiene numeri di casi approvati.
   - Se la pull request è etichettata correttamente.
7. Se le pull request soddisfano gli standard, GitHub Actions avvia il flusso di lavoro del prodotto AFT. Utilizza avvia la `ct-aft-account-request` pipeline in AWS CodePipeline. Questa pipeline avvia la macchina a stati `aft-account-provisioning-framework` personalizzata in Step Functions. Questa macchina a stati funziona come descritto in precedenza nella sezione Panoramica della soluzione di questo modello.

Implementa Account Factory for Terraform (AFT) utilizzando una pipeline bootstrap

Vinicius Elias e Edgar Costa Filho, Amazon Web Services

Riepilogo

Questo modello fornisce un metodo semplice e sicuro per implementare AWS Control Tower Account Factory for Terraform (AFT) dall'account di gestione di. AWS Organizations Il cuore della soluzione è un CloudFormation modello che automatizza la configurazione AFT creando una pipeline Terraform, strutturata per essere facilmente adattabile per la distribuzione iniziale o gli aggiornamenti successivi.

La sicurezza e l'integrità dei dati sono priorità assolute AWS, quindi il file di stato di Terraform, che è un componente fondamentale che tiene traccia dello stato dell'infrastruttura e delle configurazioni gestite, viene archiviato in modo sicuro in un bucket Amazon Simple Storage Service (Amazon S3). Questo bucket è configurato con diverse misure di sicurezza, tra cui la crittografia lato server e politiche per bloccare l'accesso pubblico, per garantire che lo stato di Terraform sia protetto da accessi non autorizzati e violazioni dei dati.

L'account di gestione orchestra e supervisiona l'intero ambiente, quindi è una risorsa fondamentale in. AWS Control Tower Questo modello segue le AWS migliori pratiche e garantisce che il processo di implementazione non sia solo efficiente, ma anche in linea con gli standard di sicurezza e governance, per offrire un modo completo, sicuro ed efficiente per implementare AFT nell'ambiente. AWS

Per ulteriori informazioni su AFT, consultate la AWS Control Tower documentazione.

Prerequisiti e limitazioni

Prerequisiti

- Un ambiente AWS multi-account di base con almeno i seguenti account: account di gestione, account Log Archive, account di audit e un account aggiuntivo per la gestione AFT.
- Un AWS Control Tower ambiente consolidato. L'account di gestione deve essere configurato correttamente, poiché il CloudFormation modello verrà distribuito al suo interno.
- Le autorizzazioni necessarie nell'account di AWS gestione. Avrai bisogno di autorizzazioni sufficienti per creare e gestire risorse come bucket S3, AWS Lambda funzioni, ruoli AWS Identity and Access Management (IAM) e progetti. AWS CodePipeline

- Familiarità con Terraform. Comprendere i concetti fondamentali e il flusso di lavoro di Terraform è importante perché l'implementazione prevede la generazione e la gestione di configurazioni Terraform.

Limitazioni

- Sii consapevole delle quote di AWS risorse nel tuo account. La distribuzione potrebbe creare più risorse e il verificarsi di quote di servizio potrebbe impedire il processo di distribuzione.
- Il modello è progettato per versioni specifiche di Terraform e. Servizi AWS L'aggiornamento o la modifica delle versioni potrebbero richiedere modifiche al modello.
- Il modello non supporta servizi di sistema di controllo della versione (VCS) autogestiti come Enterprise. GitHub

Versioni del prodotto

- Terraform versione 1.6.6 o successiva
- AFT versione 1.11 o successiva

Architecture

Stack tecnologico Target

- CloudFormation
- AWS CodeBuild
- AWS CodeCommit
- AWS CodePipeline
- Amazon EventBridge
- IAM
- AWS Lambda
- Simple Storage Service (Amazon S3)

Architettura Target

Il diagramma seguente illustra l'implementazione discussa in questo modello.

Il flusso di lavoro consiste in tre attività principali: creazione delle risorse, generazione del contenuto ed esecuzione della pipeline.

Creazione delle risorse

Il CloudFormation modello fornito con questo modello crea e imposta tutte le risorse necessarie, a seconda dei parametri selezionati al momento della distribuzione del modello. Come minimo, il modello crea le seguenti risorse:

- Una CodePipeline pipeline per implementare l'AFT
- Un bucket S3 per archiviare il file di stato Terraform associato all'implementazione AFT
- Due CodeBuild progetti per implementare il piano Terraform e applicare i comandi in diverse fasi della pipeline
- Ruoli e servizi IAM per CodeBuild CodePipeline
- Un secondo bucket S3 per archiviare gli artefatti del runtime della pipeline

A seconda del provider VCS selezionato (CodeCommit o VCS esterno), il modello crea le seguenti risorse.

- Per CodeCommit:
  - Un CodeCommit repository per archiviare il codice bootstrap AFT Terraform
  - Una EventBridge regola per acquisire le modifiche al CodeCommit repository sul ramo `main`
  - Un altro ruolo IAM per la regola EventBridge
- Per qualsiasi altro provider VCS esterno, ad esempio GitHub:
  - Una connessione AWS CodeConnections

Inoltre, quando si seleziona CodeCommit come provider VCS, se si imposta il `Generate AFT Files` parametro su`true`, il modello crea le seguenti risorse aggiuntive per generare il contenuto:

- Un bucket S3 per archiviare il contenuto generato e da utilizzare come origine del repository CodeCommit
- Una funzione Lambda per elaborare i parametri specificati e generare il contenuto appropriato
- Una funzione IAM per eseguire la funzione Lambda

- Una risorsa CloudFormation personalizzata che esegue la funzione Lambda quando il modello viene distribuito

Generazione del contenuto

Per generare i file di bootstrap AFT e il relativo contenuto, la soluzione utilizza una funzione Lambda e un bucket S3. La funzione crea una cartella nel bucket, quindi crea due file all'interno della cartella: e. `main.tf` `backend.tf` La funzione elabora anche i CloudFormation parametri forniti e compila questi file con codice predefinito, sostituendo i rispettivi valori dei parametri.

Per visualizzare il codice utilizzato come modello per generare i file, consultate l'archivio della GitHub soluzione. Fondamentalmente, i file vengono generati come segue.

principale.tf

```
module "aft" {
  source = "github.com/aws-ia/terraform-aws-control_tower_account_factory?
ref=<aft_version>"

  # Required variables
  ct_management_account_id  = "<ct_management_account_id>"
  log_archive_account_id    = "<log_archive_account_id>"
  audit_account_id          = "<audit_account_id>"
  aft_management_account_id = "<aft_management_account_id>"
  ct_home_region            = "<ct_home_region>"

  # Optional variables
  tf_backend_secondary_region = "<tf_backend_secondary_region>"
  aft_metrics_reporting       = "<false|true>"

  # AFT Feature flags
  aft_feature_cloudtrail_data_events      = "<false|true>"
  aft_feature_enterprise_support          = "<false|true>"
  aft_feature_delete_default_vpcs_enabled = "<false|true>"

  # Terraform variables
  terraform_version      = "<terraform_version>"
  terraform_distribution = "<terraform_distribution>"

  # VCS variables (if you have chosen an external VCS)
  vcs_provider                                = "<github|githubenterprise|gitlab|
gitlabselfmanaged|bitbucket>"
```

```
  account_request_repo_name                       = "<org-name>/aft-account-request"
  account_customizations_repo_name                = "<org-name>/aft-account-
customizations"
  account_provisioning_customizations_repo_name = "<org-name>/aft-account-provisioning-
customizations"
  global_customizations_repo_name                 = "<org-name>/aft-global-
customizations"

}
```

backend.ttf

```
terraform {
  backend "s3" {
    region = "<aft-main-region>"
    bucket = "<s3-bucket-name>"
    key    = "aft-setup.tfstate"
  }
}
```

Durante la creazione del CodeCommit repository, se imposti il `Generate AFT Files` parametro su`true`, il modello utilizza il bucket S3 con il contenuto generato come origine del ramo per popolare automaticamente il repository. `main`

Esecuzione della pipeline

Dopo aver creato le risorse e configurato i file di bootstrap, viene eseguita la pipeline. La prima fase (Source) recupera il codice sorgente dal ramo principale del repository e la seconda fase (Build) esegue il comando Terraform plan e genera i risultati da esaminare. Nella terza fase (Approvazione), la pipeline attende un'azione manuale per approvare o rifiutare l'ultima fase (Deploy). Nell'ultima fase, la pipeline esegue il comando Terraform utilizzando il risultato del precedente `apply` comando Terraform come input. `plan` Infine, un ruolo tra account e le autorizzazioni nell'account di gestione vengono utilizzati per creare le risorse AFT nell'account di gestione AFT.

> **Note**
>
> Se si seleziona un provider VCS esterno, sarà necessario autorizzare la connessione con le credenziali del provider VCS. Per completare la configurazione, segui i passaggi in Aggiornare una connessione in sospeso nella documentazione della AWS console Developer Tools.

Tools (Strumenti)

Servizi AWS

- CloudFormationti aiuta a configurare le risorse AWS, a fornirle in modo rapido e coerente e a gestirle durante tutto il loro ciclo di vita in tutte Account AWS le regioni.
- AWS CodeBuildè un servizio di compilazione completamente gestito che ti aiuta a compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS CodeCommitè un servizio di controllo delle versioni che consente di archiviare e gestire in modo privato gli archivi Git senza dover gestire il proprio sistema di controllo del codice sorgente.
- AWS CodePipelineconsente di modellare e configurare rapidamente le diverse fasi di un rilascio del software e di automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- AWS CodeConnectionsconsente a AWS risorse e servizi, ad esempio CodePipeline, di connettersi a repository di codice esterni, come. GitHub
- AWS Lambdaè un servizio di elaborazione che esegue il codice in risposta agli eventi e gestisce automaticamente le risorse di elaborazione, fornendo un modo rapido per creare un'applicazione di produzione moderna e senza server.
- AWS SDK per Python (Boto3)è un kit di sviluppo software che ti aiuta a integrare la tua applicazione, libreria o script Python con. Servizi AWS

Altri strumenti

- Terraform è uno strumento di infrastruttura come codice (IaC) che consente di creare, modificare e modificare l'infrastruttura in modo sicuro ed efficiente. Ciò include componenti di basso livello come istanze di calcolo, storage e rete e componenti di alto livello come voci DNS e funzionalità SaaS.
- Python è un linguaggio di programmazione potente e facile da imparare. Dispone di strutture dati efficienti di alto livello e fornisce un approccio semplice ma efficace alla programmazione orientata agli oggetti.

Archivio di codici

Il codice per questo pattern è disponibile nel repository della pipeline di bootstrap GitHub AFT.

Per il repository AFT ufficiale, vedi AWS Control Tower Account Factory for Terraform in. GitHub

Best practice

Quando distribuisci AFT utilizzando il CloudFormation modello fornito, ti consigliamo di seguire le migliori pratiche per garantire un'implementazione sicura, efficiente e di successo. Le linee guida e le raccomandazioni chiave per l'implementazione e il funzionamento dell'AFT includono quanto segue.

- Revisione approfondita dei parametri: esamina e comprendi attentamente ogni parametro del CloudFormation modello. Una configurazione accurata dei parametri è fondamentale per la corretta configurazione e funzionamento di AFT.
- Aggiornamenti regolari dei modelli: mantieni il modello aggiornato con le ultime AWS funzionalità e le versioni di Terraform. Gli aggiornamenti regolari ti aiutano a sfruttare le nuove funzionalità e a mantenere la sicurezza.
- Controllo delle versioni: aggiungi la versione del modulo AFT e, se possibile, utilizza una distribuzione AFT separata per i test.
- Ambito: utilizzate AFT solo per implementare protezioni e personalizzazioni dell'infrastruttura. Non utilizzarlo per distribuire l'applicazione.
- Linting e convalida: la pipeline AFT richiede una configurazione Terraform linkata e convalidata. Esegui lint, convalida e testa prima di inviare la configurazione ai repository AFT.
- Moduli Terraform: crea codice Terraform riutilizzabile come moduli e specifica sempre le versioni di Terraform e del AWS provider in base ai requisiti della tua organizzazione.

Epiche

Configura e configura l'ambiente AWS

Operazione	Description	Competenze richieste
Prepara l' AWS Control Tower ambiente.	Imposta e configura AWS Control Tower nel tuo AWS ambiente per garantire la gestione e la governance centralizzate per il tuo Account AWS. Per ulteriori informazioni, consulta la sezione Guida introduttiva AWS Control	Amministratore cloud

Operazione	Description	Competenze richieste
	Tower nella AWS Control Tower documentazione.	
Avvia l'account di gestione AFT.	Usa AWS Control Tower Account Factory per lanciarne uno nuovo Account AWS che funga da account di gestione AFT. Per ulteriori informazioni, consulta Fornire account con AWS Service Catalog Account Factory nella AWS Control Tower documentazione.	Amministratore cloud

Implementa il CloudFormation modello nell'account di gestione

Operazione	Description	Competenze richieste
Avvia il CloudFormation modello.	In questa epopea, distribui sci il CloudFormation modello fornito con questa soluzione per configurare la pipeline di bootstrap AFT nel tuo account di gestione. AWS La pipeline implementa la soluzione AFT nell'account di gestione AFT che hai configurato nell'epic precedente. Passaggio 1: apri la console CloudFormation • Accedi Console di gestione AWS e apri la CloudFormation console. Assicurati di operare nella regione AWS	Amministratore cloud

Operazione	Description	Competenze richieste
	Control Tower principale corretta. Fase 2: Creare un nuovo stack 1. Scegli di creare una nuova pila. 2. Seleziona l'opzione per caricare un file modello e carica il CloudFormation modello fornito con questo modello. Fase 3: Configurazione dei parametri dello stack • `VCS Provider`: Seleziona il provider del sistema di controllo della versione (VCS) da utilizzare. È possibile selezionare un VCS esterno, ad esempio GitHub, o utilizzarlo CodeCommit se l'account è autorizzato a utilizzare il servizio. • `Repository Name` : Specificate il nome del repository per la memorizza zione del modulo bootstrap AFT. Per i provider VCS esterni, utilizzate il percorso completo, incluso il nome dell'organizzazione (ad	

Operazione	Description	Competenze richieste

	esempio,). `my-github-org/my-repo`
	• `Branch Name`: Specifica re il ramo del repository di origine.
	• `CodeBuild Docker Image`: Scegli il file da usare come immagine base del CodeBuild Docker.
	• Se imposti il provider VCS su un'opzione diversa da CodeCommit, vai al passaggio 8.
	
	Passaggio 4: Decidi in che modo generare il file
	
	• Se hai selezionato CodeCommit come provider VCS, puoi utilizzare il `Generate AFT Files` parametro per controllare la generazione dei file di distribuzione AFT predefiniti. Imposta questo parametro su:
	  • `true`per creare e archiviare automatic amente i file di distribuz ione AFT nel repository specificato.
	  • `false`se desideri gestire manualmente la	

Operazione	Description	Competenze richieste

	creazione dei file o se i file sono già a posto.
	• Se hai selezionato`false`, vai al passaggio 8; in caso contrario, segui prima i passaggi da 5 a 7.
	
	Passaggio 5: inserisci i dettagli dell' AWS Control Tower account AFT
	
	• Se imposti il `Generate AFT Files` parametro su`true`, fornisci le seguenti informazioni AWS Control Tower e quelle specifiche dell'account AFT.
	  • `Log Archive Account ID`: L'ID dell'ID dell'account Log Archive in. AWS Control Tower
	  • `Audit Account ID`: L'ID dell'account di controllo in AWS Control Tower.
	  • `AFT Management Account ID`: L'ID dell'account di gestione AFT che hai creato nella prima epic.
	  • `AFT Main Region`AFT `Secondary Region`: principale e secondari	

Operazione	Description	Competenze richieste
	o Regioni AWS per l'implementazione di AFT. Fase 6: Configurazione delle opzioni AFT • Imposta la reportistica delle metriche: • `AFT Enable Metrics Reporting` : abilita o disabilita il reporting delle metriche AFT. Per ulteriori informazioni, consulta Metriche operative nella AWS Control Tower documentazione. • Imposta le opzioni della funzionalità AFT: • `Enable AFT CloudTrail Data Events`: abilita gli eventi CloudTrail relativi ai dati in tutti gli account gestiti da AFT. Per ulteriori informazioni, consulta gli eventi AWS CloudTrail relativi ai dati nella AWS Control Tower documenta zione. • `Enable AFT Enterprise Support` : Abilita Enterprise Support in tutti gli account	

Operazione	Description	Competenze richieste
	gestiti AFT. Per ulteriori informazioni, consulta il piano AWS Enterprise Support nella AWS Control Tower documentazione. • `Enable AFT Delete Default VPC`: Elimina tutto solo VPCs nell'account di gestione AFT. Per ulteriori informazioni, consulta Eliminare il VPC AWS predefinito nella AWS Control Tower documentazione. Passaggio 7: Specificare le versioni • `AFT Terraform Version`: Scegli la versione di Terraform da utilizzare nelle pipeline AFT. • `AFT Version`: Definisci la versione AFT per la distribuzione. Mantenete l'impostazione predefinita (`latest`) per utilizzare la versione AFT più recente. Fase 8: Rivedi e crea lo stack • Rivedi tutti i parametri e le impostazioni. Se tutto è	

Operazione	Description	Competenze richieste
	in ordine, procedi con la creazione dello stack. Passaggio 9: Monitora la creazione dello stack • CloudFormation effettua il provisioning e configura le risorse che hai definito. Monitora il processo di creazione dello stack sulla CloudFormation console. Questo processo potrebbe richiedere alcuni minuti. Passaggio 10: verifica della distribuzione • Quando lo stato dello stack mostra CREATE_COMPLETE, verificate che tutte le risorse siano state create correttamente. • Nella sezione Output, annota il valore. `TerraformBackendBucketName`	

Compila e convalida il repository e la pipeline di bootstrap AFT

Operazione	Description	Competenze richieste
Opzione 1: popolare il repository bootstrap AFT per un VCS esterno.	Se impostate il provider VCS su un VCS esterno (non su), seguite questi passaggi. CodeCommit (Facoltativo) Dopo aver distribuito il CloudFormation modello, potete compilare o convalidare il contenuto nel repository bootstrap AFT appena creato e verificare se la pipeline è stata eseguita correttamente. Fase 1: Aggiornare la connessione 1. Sulla CodePipeline console, nel riquadro di navigazione, scegli Impostazioni, Connessioni. 2. Seleziona la `aft-vcs-connection` connessione. Dovrebbe essere in `Pending` stato. 3. Scegli Aggiorna connessione in sospeso e segui le istruzioni in Aggiornare una connessione in sospeso nella documentazione della console Developer Tools.	Amministratore cloud

Operazione	Description	Competenze richieste
	4. Quando la connessione è in `Available` stato, vai al passaggio successivo. Passaggio 2: popolare il repository 1. Usa le tue credenziali VCS esterne per clonare il repository specificato nel modello sul tuo computer locale. Se hai mantenuto il nome predefinito, viene chiamato il repository. `aft-setup` 2. Nel repository, crea una cartella chiamata `terraform` con due file vuoti al suo interno: `backend.tf` e. `main.tf` 3. Apri il `backend.tf` file e aggiungi questo frammento di codice: `terraform { backend "s3" { region = "<aft-main-region>" bucket = "<s3-bucket-name>" key = "aft-setup" } }` Nel file:	

Operazione	Description	Competenze richieste
	• Sostituisci `<aft-main-region>` con la regione AFT principale. Dovrebbe corrispondere alla regione AWS Control Tower principale. • Sostituisci `<s3-bucket-name>` con il nome del bucket di backend Terraform. Puoi trovarlo nell'`TerraformBackendBucketName` output generato dal CloudFormation modello che hai distribuito in precedenza. 4. Apri il `main.tf` file e usa uno degli esempi disponibili nel repository AFT per distribuire AFT. Ad esempio, puoi lavorare con il tuo provider VCS preferito (CodeCommito Bitbucket) o personalizzare il VPC AFT. GitHub Per ulteriori opzioni di input AFT, consultate il file README nell'archivio AFT. Fase 2: Conferma e invia le modifiche • Dopo aver creato e compilato la cartella e i file,	

Operazione	Description	Competenze richieste
	conferma le modifiche e carica il codice nel repository. La pipeline si avvia automaticamente, attraversa le fasi Source e Build, quindi attende un'azione di approvazione prima della fase di distribuzione.	

Operazione	Description	Competenze richieste
Opzione 2: popolare il repository bootstrap AFT per. CodeCommit	Se imposti il provider VCS su CodeCommit, segui questi passaggi. (Facoltativo) Dopo aver distribuito il CloudFormation modello, potete compilare o convalidare il contenuto nel repository bootstrap AFT appena creato e verificare se la pipeline è stata eseguita correttamente. Se impostate il `Generate AFT Files` parametro su`true`, passate alla storia successiva (convalida della pipeline). Passaggio 1: popolare il repository 1. Apri la AWS CodeCommit console e seleziona il repository appena creato. Se hai mantenuto il nome predefinito, il nome del repository dovrebbe essere. `aft-setup` 2. Clona il repository sul tuo computer locale usando SSH, HTTPS o HTTPS (GRC) e aprilo in un editor. 3. Crea una cartella chiamata `terraform` e due file	Amministratore cloud

Operazione	Description	Competenze richieste
	vuoti al suo interno: e. `backend.tf main.tf` 4. Apri il `backend.tf` file e aggiungi questo frammento di codice: `terraform { backend "s3" { region = "<aft-main-region>" bucket = "<s3-bucket-name>" key = "aft-setup" } }` Nel file: • Sostituisci `<aft-main-region>` con la regione AFT principale. Dovrebbe corrispondere alla regione AWS Control Tower principale. • Sostituisci `<s3-bucket-name>` con il nome del bucket di backend Terraform. Puoi trovarlo nell'`TerraformBackendBucketName` output generato dal CloudFormation modello che hai distribuito in precedenza.	

Operazione	Description	Competenze richieste
	5. Apri il `main.tf` file e usa uno degli esempi disponibili nel repository AFT per distribuire AFT. Ad esempio, puoi lavorare con il tuo provider di sistema di controllo della versione (VCS) preferito (CodeCommito Bitbucket) o personalizzare il VPC AFT. GitHub Per ulteriori opzioni di input AFT, consultate il file README nell'archivio AFT. Fase 2: Conferma e invia le modifiche • Dopo aver creato e compilato la cartella e i file, conferma le modifiche e carica il codice nel repository. La pipeline si avvia automaticamente, attraversa le fasi Source e Build, quindi attende un'azione di approvazione prima della fase di distribuzione.	

Operazione	Description	Competenze richieste
Convalida la pipeline di bootstrap AFT.	Fase 1: Visualizza la pipeline • Apri la CodePipeline console e verifica se la `aft-bootstrap-pipeline` pipeline è stata avviata correttamente. Dovrebbe eseguire un piano Terraform o attendere un'azione di approvazione manuale. Passaggio 2: approvare i risultati del piano Terraform • È possibile esaminare i risultati del piano Terraform esaminando i registri di esecuzione della fase di compilazione, quindi approvare o rifiutare l'esecuzione nella fase di approvazione. Se approvi, la pipeline inizia a distribuire le risorse AFT nell'account di gestione AFT fornito. Fase 3: Attendi la distribuzione • Attendi che la pipeline funzioni correttamente. Questa operazione dovrebbe richiedere circa 30 minuti. Gli eventuali errori che potresti riscontrare sono	Amministratore cloud

Operazione	Description	Competenze richieste
	spesso causati dalle quote API. In questi casi, puoi eseguire nuovamente la pipeline per continuare la distribuzione. Fase 4: Controlla le risorse create • Accedi all'account di gestione AFT e conferma che le risorse sono state create.	

risoluzione dei problemi

Problema	Soluzione
La funzione Lambda personalizzata inclusa nel CloudFormation modello non funziona durante la distribuzione.	Controlla i CloudWatch log di Amazon per la funzione Lambda per identificare l'errore. I log forniscono informazioni dettagliate e possono aiutare a individuare il problema specifico. Verifica che la funzione Lambda disponga delle autorizzazioni necessarie e che le variabili di ambiente siano state impostate correttamente.
Si verificano errori nella creazione o nella gestione delle risorse causati da autorizzazioni inadeguate.	Esamina i ruoli e le policy IAM collegati alla funzione Lambda e altri servizi coinvolti nella distribuzione. CodeBuild Verifica che dispongan o delle autorizzazioni necessarie. In caso di problemi di autorizzazione, modifica le politiche IAM per concedere l'accesso richiesto.

Problema	Soluzione
Stai utilizzando una versione obsoleta del CloudFormation modello con versioni più recenti Servizi AWS o Terraform.	Aggiorna regolarmente il CloudFormation modello per renderlo compatibile con le ultime versioni AWS e con le versioni di Terraform. Consulta le note di rilascio o la documentazione per eventuali modifiche o requisiti specifici della versione.
Le Servizio AWS quote vengono raggiunte durante la distribuzione.	Prima di distribuire la pipeline, controlla le Servizio AWS quote per risorse come i bucket S3, i ruoli IAM e le funzioni Lambda. La richiesta aumenta se necessario. Per ulteriori informazioni, consulta le Servizio AWS quote sul AWS sito Web.
Si verificano errori dovuti a parametri di input errati nel CloudFormation modello.	Ricontrolla tutti i parametri di input per errori di battitura o valori errati. Verifica che gli identific atori delle risorse, come i nomi degli account IDs e delle regioni, siano accurati.

Risorse correlate

Per implementare correttamente questo modello, consulta le seguenti risorse. Queste risorse forniscono informazioni e indicazioni aggiuntive che possono essere preziose per la configurazione e la gestione di AFT utilizzando CloudFormation.

AWSdocumentazione:

- AWS Control Tower La Guida per l'utente offre informazioni dettagliate sulla configurazione e la gestione AWS Control Tower.
- CloudFormation la documentazione fornisce approfondimenti su CloudFormation modelli, stack e gestione delle risorse.

Politiche e best practice IAM:

- Le best practice di sicurezza in IAM spiegano come contribuire a proteggere AWS le risorse utilizzando i ruoli e le policy IAM.

Terraform su AWS:

- La documentazione di Terraform AWS Provider fornisce informazioni complete sull'utilizzo di Terraform con. AWS

Servizio AWS quote:

- Servizio AWS le quote forniscono informazioni su come visualizzare le Servizio AWS quote e su come richiedere aumenti.

Gestisci i prodotti AWS Service Catalog in più account AWS e regioni AWS

Ram Kandaswamy, Amazon Web Services

Riepilogo

Amazon Web Services (AWS) Service Catalog semplifica e accelera la governance e la distribuzione dei modelli Infrastructure as Code (IaC) per le aziende. CloudFormation I modelli AWS vengono utilizzati per definire una raccolta di risorse AWS (stack) necessarie per un prodotto. AWS CloudFormation StackSets estende questa funzionalità consentendoti di creare, aggiornare o eliminare stack su più account e regioni AWS con un'unica operazione.

Gli amministratori di AWS Service Catalog creano prodotti utilizzando CloudFormation modelli creati dagli sviluppatori e li pubblicano. Questi prodotti vengono quindi associati a un portafoglio e vengono applicati dei vincoli per la governance. Per rendere i tuoi prodotti disponibili agli utenti in altri account o unità organizzative AWS (OUs), in genere condividi il tuo portafoglio con loro. Questo modello descrive un approccio alternativo per la gestione delle offerte di prodotti AWS Service Catalog basato su AWS CloudFormation StackSets. Invece di condividere i portafogli, utilizzi i vincoli dello stack set per impostare le regioni e gli account AWS in cui il prodotto può essere distribuito e utilizzato. Utilizzando questo approccio, puoi effettuare il provisioning dei prodotti AWS Service Catalog su più account e regioni AWS e gestirli da una posizione centrale, soddisfacendo al contempo i requisiti di governance. OUs

Vantaggi di questo approccio:

- Il prodotto viene fornito e gestito dall'account principale e non viene condiviso con altri account.
- Questo approccio fornisce una visione consolidata di tutti i prodotti (pile) forniti e basati su un prodotto specifico.
- La configurazione con AWS Service Management Connector è più semplice, perché si rivolge a un solo account.
- È più facile interrogare e utilizzare i prodotti di AWS Service Catalog.

Prerequisiti e limitazioni

Prerequisiti

- CloudFormation Modelli AWS per IaC e controllo delle versioni

- Configurazione di più account e AWS Service Catalog per il provisioning e la gestione delle risorse AWS

Limitazioni

- Questo approccio utilizza AWS CloudFormation StackSets e StackSets si applicano le seguenti limitazioni:
  - StackSets non supporta la distribuzione di CloudFormation modelli tramite macro. Se utilizzi una macro per preelaborare il modello, non sarai in grado di utilizzare una distribuzione StackSets basata.
  - StackSets offre la possibilità di dissociare uno stack dal set di stack, in modo da poter scegliere come destinazione uno stack specifico per risolvere un problema. Tuttavia, uno stack dissociato non può essere riassociato allo stack set.
- AWS Service Catalog genera automaticamente StackSet i nomi. La personalizzazione non è attualmente supportata.

Architecture

Architettura di Target

1. L'utente crea un CloudFormation modello AWS per il provisioning delle risorse AWS, in formato JSON o YAML.
2. Il CloudFormation modello crea un prodotto in AWS Service Catalog, che viene aggiunto a un portafoglio.
3. L'utente crea un prodotto fornito, che crea CloudFormation pile negli account di destinazione.
4. Ogni stack fornisce le risorse specificate nei modelli. CloudFormation

Tools (Strumenti)

Servizi AWS

- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account e regioni AWS.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.

- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Service Catalog ti aiuta a gestire centralmente i cataloghi di servizi IT approvati per AWS. Gli utenti finali possono distribuire rapidamente soltanto i servizi IT approvati di cui hanno bisogno, in accordo con i vincoli stabiliti dall'organizzazione.

Epiche

Fornisci prodotti su più account

Operazione	Description	Competenze richieste
Crea un portfolio.	Un portafoglio è un contenitore che include uno o più prodotti raggruppati in base a criteri specifici. L'utilizzo di un portafoglio per i tuoi prodotti ti aiuta ad applicare vincoli comuni a tutti i tuoi set di prodotti. Per creare un portfolio, segui le istruzioni nella documentazione di AWS Service Catalog. Se utilizzi la CLI di AWS, ecco un comando di esempio: `aws servicecatalog create-portfolio --provider-name my-provider --display-name my-portfolio` Per ulteriori informazioni, consulta la documentazione dell'interfaccia a riga di comando di AWS.	AWS Service Catalog, IAM

Operazione	Description	Competenze richieste
Crea un CloudFormation modello.	Crea un CloudFormation modello che descriva le risorse. I valori delle proprietà delle risorse devono essere parametrizzati ove applicabile.	AWS CloudFormation, JSON/YAML

Operazione	Description	Competenze richieste
Crea un prodotto con informazioni sulla versione.	Il CloudFormation modello diventa un prodotto quando viene pubblicato in AWS Service Catalog. Fornisci valori per i parametri opzionali di dettaglio della versione, come il titolo e la descrizione della versione; ciò ti sarà utile per richiedere informazioni sul prodotto in un secondo momento. Per creare un prodotto, segui le istruzioni nella documentazione di AWS Service Catalog. Se utilizzi la CLI di AWS, un comando di esempio è: `aws servicecatalog create-product --cli-input-json file://create-product-input.json` dove `create-product-input.json` è il file che passa i parametri per il prodotto. Per un esempio di questo file, consulta la sezione Informazioni aggiuntive. Per ulteriori informazioni, consulta la documentazione dell'interfaccia a riga di comando di AWS.	AWS Service Catalog

Operazione	Description	Competenze richieste
Applica vincoli.	Applica i vincoli dello stack set al portafoglio, per configura re opzioni di distribuzione del prodotto come più account AWS, regioni e autorizza zioni. Per istruzioni, consulta la documentazione di AWS Service Catalog.	AWS Service Catalog
Aggiungi autorizzazioni	Fornisci le autorizzazioni agli utenti in modo che possano lanciare i prodotti del portafogl io. Per istruzioni sulla console, consulta la documentazione di AWS Service Catalog. Se utilizzi la CLI di AWS, ecco un comando di esempio: `aws servicecatalog associate-principal-with-portfolio \ --portfolio-id port-2s6abcdefwdh4 \ --principal-arn arn:aws:iam::444455556666:role/Admin \ --principal-type IAM` Per ulteriori informazioni, consulta la documentazione dell'interfaccia a riga di comando di AWS.	AWS Service Catalog, IAM

Operazione	Description	Competenze richieste
Effettua il provisioning del prodotto.	Un prodotto sottoposto a provisioning è un'istanza di risorse di un prodotto. Il provisioning di un prodotto basato su un CloudFormation modello avvia uno CloudFormation stack e le relative risorse sottostanti. Effettua il provisioning del prodotto scegliendo come target le regioni e gli account AWS applicabili, in base ai vincoli dello stack set. Nella CLI di AWS, ecco un comando di esempio: `aws servicecatalog provision-product \ --product-id prod-abcdfz3syn2rg \ --provisioning-artifact-id pa-abc347pcsccfm \ --provisioned-product-name "mytestppname3"` Per ulteriori informazioni, consulta la documentazione dell'interfaccia a riga di comando di AWS.	AWS Service Catalog

Risorse correlate

Riferimenti

- Panoramica di AWS Service Catalog
- Usare AWS CloudFormation StackSets

Tutorial e video

- AWS re:Invent 2019: Automatizza tutto: opzioni e best practice (video)

Informazioni aggiuntive

Quando usi il `create-product` comando, il `cli-input-json` parametro punta a un file che specifica informazioni come il proprietario del prodotto, l'e-mail di supporto e i dettagli del modello. CloudFormation Ecco un esempio di tale file:

```
{
   "Owner": "Test admin",
      "SupportDescription": "Testing",
         "Name": "SNS",
            "SupportEmail": "example@example.com",
            "ProductType": "CLOUD_FORMATION_TEMPLATE",
               "AcceptLanguage": "en",
                  "ProvisioningArtifactParameters": {
                     "Description": "SNS product",
                        "DisableTemplateValidation": true,
                           "Info": {
                              "LoadTemplateFromURL": "<url>"
                     },
                           "Name": "version 1"
}
```

Monitora i cluster SAP RHEL Pacemaker utilizzando i servizi AWS

Harsh Thoria, Randy Germann e RAVEENDRA Voore, Amazon Web Services

Riepilogo

Questo modello delinea i passaggi per il monitoraggio e la configurazione degli avvisi per un cluster Red Hat Enterprise Linux (RHEL) Pacemaker per applicazioni SAP e servizi di database SAP HANA utilizzando Amazon e Amazon Simple Notification Service (Amazon CloudWatch SNS).

La configurazione consente di monitorare le risorse del cluster SAP SCS o ASCS, Enqueue Replication Server (ERS) e SAP HANA quando si trovano in uno stato «interrotto» con l'aiuto di flussi di CloudWatch log, filtri metrici e allarmi. Amazon SNS invia un'e-mail all'infrastruttura o al team SAP Basis sullo stato del cluster interrotto.

È possibile creare le AWS risorse per questo modello utilizzando AWS CloudFormation gli script o le AWS console di servizio. Questo modello presuppone che tu stia utilizzando le console; non fornisce CloudFormation script né copre la distribuzione dell'infrastruttura per Amazon CloudWatch SNS. I comandi Pacemaker vengono utilizzati per impostare la configurazione degli avvisi del cluster.

Prerequisiti e limitazioni

Prerequisiti

- Un account attivo. AWS
- Amazon SNS configurato per inviare notifiche e-mail o mobili.
- Un cluster SAP ASCS/ERS per ABAP o SCS/ERS per Java e un cluster RHEL Pacemaker del database SAP HANA. Per le istruzioni, consulta quanto segue:
  - Configurazione del cluster SAP HANA
  - Configurazione del cluster SAP Netweaver ABAP/Java

Limitazioni

- Questa soluzione attualmente funziona con i cluster basati su Pacemaker versione 7.3 e successive di RHEL. Non è stata testata sui sistemi operativi SUSE.

Versioni del prodotto

- RHEL 7.3 e versioni successive

Architecture

Stack tecnologico Target

- Agente basato sugli eventi di avviso RHEL Pacemaker
- Amazon Elastic Compute Cloud (Amazon EC2)
- CloudWatch allarme
- CloudWatch gruppo di log e filtro metrico
- Amazon SNS

Architettura di Target

Il diagramma seguente illustra i componenti e i flussi di lavoro di questa soluzione.

Automazione e scalabilità

- È possibile automatizzare la creazione di AWS risorse utilizzando CloudFormation gli script. Puoi anche utilizzare filtri metrici aggiuntivi per ridimensionare e coprire più cluster.

Tools (Strumenti)

Servizi AWS

- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue AWS risorse e delle applicazioni su cui esegui AWS in tempo reale.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.

Strumenti

- CloudWatch agent (unified) è uno strumento che raccoglie parametri, log e tracce a livello di sistema dalle EC2 istanze e recupera metriche personalizzate dalle tue applicazioni.
- Pacemaker alert agent (per RHEL 7.3 e versioni successive) è uno strumento che avvia un'azione in caso di modifica, ad esempio quando una risorsa si arresta o si riavvia, in un cluster Pacemaker.

Best practice

- Per le best practice per l'utilizzo dei carichi di lavoro SAP su AWS, consulta SAP Lens for the Well-Architected AWS Framework.
- Considera i costi associati alla configurazione del CloudWatch monitoraggio per i cluster SAP HANA. Per ulteriori informazioni, consulta la documentazione relativa ad CloudWatch .
- Valuta la possibilità di utilizzare un cercapersone o un meccanismo di ticketing per gli avvisi di Amazon SNS.
- Controlla sempre le versioni RHEL ad alta disponibilità (HA) del pacchetto RPM per PC, Pacemaker e Fencing Agent. AWS

Epiche

Configurazione di Amazon SNS

Operazione	Description	Competenze richieste
Creare un argomento SNS.	1. Accedi Console di gestione AWS e apri la console Amazon SNS nella versione v3/home. https://console.aws.amazon.com/sns/ 2. Sul pannello di controllo Amazon SNS sotto Common actions (Operazioni comuni), scegli Create Topic (Crea argomento). 3. Nella finestra di dialogo Crea nuovo argomento, per Tipo, scegli Standard. 4. Per Nome argomento, inserite un nome per l'argomento (ad esempio,`my-topic`).	Amministratore AWS

Operazione	Description	Competenze richieste
	5. Scegli Create topic (Crea argomento). In questo modo viene creato un argomento SNS con una politica delle risorse che consente di pubblicare notifiche. 6. Copia l'ARN dell'argomento (ad esempio,`arn:aws:sns:us-east-1:111122223333:my-topic`). Utilizzerai questo ARN in una fase successiva.	

Operazione	Description	Competenze richieste
Modifica la politica di accesso per l'argomento SNS.	1. Sulla console Amazon SNS, nel pannello di navigazione, scegli Argomenti, quindi scegli l'argomento che hai creato. 2. Scegli Modifica e vai alla sezione Politica di accesso. 3. Assicurati che la politica di accesso CloudWatch includa tra i principali servizi autorizzati a pubblicare su questo argomento. Esempio: `{ "Sid": "Allow AWS CloudWatch to Publish to this SNS topic", "Effect": "Allow", "Principal": { "Service": [ "cloudwatch.amazonaws.com" ] }, "Action": "SNS:Publish", "Resource": "arn:aws:sns:us-east-1:111122223333:my-topic" }` 4. Scegli Save changes (Salva modifiche).	Amministratore di sistema AWS

Operazione	Description	Competenze richieste
Iscriviti all'argomento SNS.	1. Sulla console Amazon SNS, nel pannello di navigazione, scegli Abbonamenti, Crea abbonamento. 2. Per Argomento ARN, incolla l'ARN che hai creato nella prima attività. 3. Per Protocollo, scegli E-mail. 4. Per Endpoint, inserisci l'indirizzo e-mail della persona o del team responsabile del cluster SAP Pacemaker e che deve ricevere le notifiche . Ad esempio, può essere l'indirizzo e-mail per la lista di distribuzione di SAP Basis o del team di infrastruttura. 5. Scegli Create Subscription (Crea sottoscrizione). 6. Dalla tua applicazione di posta elettronica, apri il messaggio da AWS Notifiche e conferma l'iscrizione. Nel Web browser viene visualizzata una risposta di conferma di Amazon SNS.	Amministratore di sistema AWS

Conferma la configurazione del cluster

Operazione	Description	Competenze richieste
Controlla lo stato del cluster.	Usa il comando pcs status per confermare che le risorse sono online.	Amministratore SAP Basis

Configura gli avvisi Pacemaker

Operazione	Description	Competenze richieste

Configurare l'agente di avviso Pacemaker sull'istanza del cluster principale.	Accedi all' EC2 istanza nel cluster primario ed esegui i seguenti comandi:
	`install --mode=0755 /usr/share/pacemaker/alerts/alert_file.sh.sample`
	`touch /var/lib/pacemaker/alert_file.sh`
	`touch /var/log/pcmk_alert_file.log`
	`chown hacluster:haclient /var/log/pcmk_alert_file.log`
	`chmod 600 /var/log/pcmk_alert_file.log`
	`pcs alert create id=alert_file description="Log events to a file." path=/var/lib/pacemaker/alert_file.sh`
	`pcs alert recipient add alert_file id=my-alert_logfile value=/va`
		Amministratore SAP Basis

Operazione	Description	Competenze richieste
	`r/log/pcmk_alert_f ile.log`	

Configurare l'agente di avviso Pacemaker sull'istanza del cluster secondaria.	Accedi all' EC2 istanza del cluster secondario nel cluster secondario ed esegui i seguenti comandi:
	`install --mode=0755 / usr/share/pacemak er/alerts/alert_fi le.sh.sample`
	`touch /var/lib/ pacemaker/alert_fi le.sh`
	`touch /var/log/ pcmk_alert_file.log`
	`chown hacluster :haclient /var/log/ pcmk_alert_file.log`
	`chmod 600 /var/log/ pcmk_alert_file.log`
		Amministratore SAP Basis

Operazione	Description	Competenze richieste

Conferma che la risorsa di avviso RHEL è stata creata.	Utilizzate il seguente comando per confermare che la risorsa di avviso è stata creata:
	`pcs alert`
	L'output del comando sarà simile al seguente:
	`[root@xxxxxxx ~]# pcs alert`
	`Alerts:`
	`Alert: alert_file (path=/var/lib/pacemaker/alert_file.sh)`
	`Description: Log events to a file.`
	`Recipients:`
	`Recipient: my-alert_logfile (value=/var/log/pcmk_alert_file.log)`
		Amministratore SAP Basis

Configurare l'agente CloudWatch

Operazione	Description	Competenze richieste
Installa l' CloudWatch agente.	Esistono diversi modi per installare l' CloudWatch agente su un' EC2 istanza. Per utilizzare la riga di comando: 1. Scarica il pacchetto CloudWatch dell'agente:	Amministratore di sistema AWS

Operazione	Description	Competenze richieste

	`wget https://s3.<region>.amazonaws.com/amazoncloudwatch-agent-region/redhat/amd64/latest/amazon-cloudwatch-agent.rpm`
	
	`<region>`dov'è la posizione Regione AWS in cui si trova l' EC2 istanza (ad esempio,`us-west-2`).
	2. Facoltativo) Verifica la firma del pacchetto. Per istruzioni, consulta Verifica della firma del pacchetto dell' CloudWatch agente nella CloudWatch documentazione.
	3. Installa il pacchetto in prima istanza:
	
	`sudo rpm -U ./amazon-cloudwatch-agent.rpm`
	4. Ripetere l'operazione per l'istanza secondaria.
	
	Per ulteriori informazioni, consulta la documentazione relativa ad CloudWatch .	

Operazione	Description	Competenze richieste
Associa un ruolo IAM all' EC2 istanza.	Per consentire all' CloudWatch agente di inviare dati dalle istanze, devi associare il CloudWatchAgentServerRoleruolo IAM a ciascuna istanza. In alternativa, puoi aggiungere una policy per l' CloudWatch agente al tuo ruolo IAM esistente. Per ulteriori informazioni, consulta la documentazione relativa ad CloudWatch .	Amministratore AWS

Operazione	Description	Competenze richieste
Configura l' CloudWatch agente per monitorare il file di registro dell'agente di avviso Pacemaker sull'istanza del cluster principale.	1. Configura l'istanza del cluster principale eseguendo il comando: `sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-config-wizard` 2. Scegli 1 per Linux, quindi seleziona le opzioni per la tua strategia di monitoraggio. 3. Per la domanda «Vuoi monitorare qualsiasi file di registro», scegli Sì e fornisci il percorso del file di registro di Pacemaker dal comando pcs alert. Nel nostro caso, lo è. `var/log/pcmk_alert_file.log` 4. Fornisci il nome del gruppo di log e del flusso di log. Se non si specifica un flusso di log, l'ID dell' AWS istanza viene utilizzato come impostazione predefinita. 5. Ripetere i passaggi 1-4 per l'istanza del cluster secondaria.	Amministratore AWS

Operazione	Description	Competenze richieste
Avvia l' CloudWatch agente sulle istanze del cluster primario e secondario.	Per avviare l'agente, esegui il comando seguente sulle EC2 istanze nei cluster primario e secondario: `sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m ec2 -s -c file:/opt/aws/amazon-cloudwatch-agent/bin/config.json`	Amministratore AWS

Configura CloudWatch le risorse

Operazione	Description	Competenze richieste
Configura gruppi di CloudWatch log.	1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/ 2. Nel riquadro di navigazione, scegli Gruppi di log, Crea gruppo di log. 3. Inserisci un nome per il gruppo di log, quindi scegli Crea gruppo di log. L' CloudWatch agente trasferirà il file di avviso Pacemaker al gruppo di CloudWatch log come flusso di log.	Amministratore AWS

Operazione	Description	Competenze richieste
Configura filtri CloudWatch metrici.	I filtri metrici ti aiutano a cercare uno schema, ad esempio `stop <cluster-resource-name>` nei flussi di CloudWatch log. Quando viene identificato questo modello, il filtro metrico aggiorna una metrica personalizzata. 1. Sulla CloudWatch console, nel riquadro di navigazione, scegli Registra gruppi. 2. Scegli il nome del gruppo di log che hai creato nell'atti vità precedente. 3. Scegliere Operazioni, Crea filtro parametri. 4. Per Filter pattern, inserisci il pattern di filtro da utilizzar e, ad esempio`stop ABC_scs`, per abbinare l'evento di stop per una risorsa del cluster SAP SCS denominata. `ABC_scs` Per ulteriori informazi oni, consulta la sintassi del pattern di filtro nella documentazione. CloudWatch 5. (Facoltativo) Per testare il modello di filtro, in Modello di test, inserisci uno o più	Amministratore AWS, amministratore SAP Basis

Operazione	Description	Competenze richieste
	log eventi da utilizzare per testare il modello. Ogni evento di registro deve essere specificato su una riga separata, poiché le interruzioni di riga vengono utilizzate per separare gli eventi di registro nella casella Registra i messaggi degli eventi. 6. Scegli Next (Successivo) e poi inserisci un nome per il filtro. 7. In Dettagli metrici, per Metric namespace, inserisci un nome per lo spazio dei CloudWatch nomi in cui verrà pubblicata la metrica (ad esempio,). `sapcluster_monitoring` Se questo spazio dei nomi non esiste già, seleziona Crea nuovo. 8. Per Nome della metrica, inserisci un nome per la nuova metrica (ad esempio`sapcluster_<sid>` , `<sid>` dov'è il nome di identificazione del sistema SAP). 9. Per Valore metrico, inserisci 1.	

Operazione	Description	Competenze richieste
	In alternativa, puoi inserire un token come`$size`. In questo modo il parametro viene incrementato in base al valore del numero nel campo `size` per ogni log eventi che contiene un campo `size`. 10. Per Valore predefinito, inserisci 0. 11. Scegli Crea filtro parametri. Quando il filtro metrico identifica il modello nel passaggio 4, aggiorna il valore della metrica CloudWatch **`sapcluster_abc`** personalizzata a 1. L' CloudWatch allarme **`SAP-Cluster-QA1-ABC`** monitora la metrica **`sapcluster_abc`** e invia una notifica SNS quando il valore della metrica passa a 1. Ciò indica che la risorsa del cluster si è interrotta e che è necessario intraprendere un'azione.	

Operazione	Description	Competenze richieste
Imposta un allarme CloudWatch metrico per la metrica SAP ASCS/SCS ed ERS.	Per creare un allarme basato su una singola metrica: 1. Sulla CloudWatch console, nel pannello di navigazione, scegli Allarmi, Tutti gli allarmi. 2. Scegli Crea allarme. 3. Scegli Select Metric (Seleziona parametro). 4. Cerca la metrica `sapcluster_monitoring` personalizzata creata nell'attività precedente. 5. Scegli il nome della metrica per SAP SCS (ad esempio,`sapcluster_<abc>`), anch'esso creato nell'attività precedente. 6. Nella scheda Metriche grafiche, imposta quanto segue: • Per Statistic (Statistica), scegli Maximum (Massima). • Per Periodo, scegli 1 minuto. • Per Tipo di soglia, scegliete Statico e impostate la soglia **`sapcluste`**	Amministratore AWS

Operazione	Description	Competenze richieste
	r_<sid> per un valore maggiore o uguale a 1. 7. Scegli Next (Successivo). 8. Per Notifica, seleziona l'argomento SNS che hai creato nella prima epic. 9. Per Nome e descrizione, fornisci il nome dell'allarme e una breve descrizione, quindi scegli Avanti. 10. Scegli Crea allarme.	
Imposta un allarme CloudWatch metrico per la metrica SAP HANA.	Ripeti i passaggi per impostare un allarme CloudWatch metrico dell'attività precedente, con queste modifiche: • Per il passaggio 5, scegli il nome della metrica per SAP HANA (ad esempio,). `sapcluster_db_<abc>` • Per il passaggio 6, imposta la soglia `sapcluster_<sid>` per un valore maggiore di 0.	Amministratore AWS

Risorse correlate

- Attivazione di script per eventi cluster (documentazione RHEL)
- Creare il file di configurazione CloudWatch dell'agente con la procedura guidata (documentazione) CloudWatch
- Installazione ed esecuzione dell' CloudWatch agente sui server (CloudWatch documentazione)

- Crea un CloudWatch allarme basato su una soglia statica (CloudWatch documentazione)
- Distribuzione manuale di SAP HANA su AWS con cluster ad alta disponibilità (documentazione SAP sul sito Web) AWS
- NetWeaver Guide SAP (documentazione SAP sul sito Web) AWS

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Monitora l'attività delle applicazioni utilizzando CloudWatch Logs Insights

Ram Kandaswamy, Amazon Web Services

Riepilogo

Questo modello fornisce una soluzione per rilevare e inviare avvisi automaticamente sulle eccezioni delle applicazioni utilizzando Amazon CloudWatch Logs Insights. Implementando l'analisi e l'invio di avvisi automatici dei log, puoi identificare e rispondere rapidamente ai problemi delle applicazioni nel tuo ambiente di produzione.

I log svolgono un ruolo cruciale nel monitoraggio del comportamento del sistema, nell'identificazione dei problemi e nel garantire prestazioni ottimali. Durante un processo di migrazione, i file di registro sono preziosi per convalidare il funzionamento del sistema nel nuovo ambiente, rilevare problemi di compatibilità e identificare eventuali comportamenti imprevisti. I problemi potrebbero essere correlati alle operazioni o alla sicurezza. Per quanto riguarda i problemi relativi alla sicurezza, consentire il rilevamento tempestivo di tentativi di accesso non autorizzati o attività sospette è essenziale per mantenere la sicurezza e la conformità normativa. Questa funzionalità è particolarmente importante quando si tratta di dati sensibili o sistemi critici.

Questo modello è particolarmente utile per i team che devono eseguire le seguenti operazioni:

- Mantieni un'elevata disponibilità delle applicazioni.
- Rispondi rapidamente ai problemi di produzione.
- Analizza gli errori specifici dell'applicazione non rilevati dai log. Servizio AWS
- Esegui l'analisi dei log su richiesta senza un'infrastruttura precostruita.

CloudWatch Logs Insights è ottimale per analizzare i log generati dalle applicazioni in cui il contesto dell'errore esiste solo all'interno del codice dell'applicazione. CloudWatch Logs Insights eccelle nelle seguenti attività:

- Interroga dati di log non strutturati o semistrutturati.
- Esegui analisi su richiesta durante la risposta agli incidenti.
- Correla gli eventi su più gruppi di log.
- Crea visualizzazioni rapide senza strumenti esterni.

Prerequisiti e limitazioni

Prerequisiti

- Un'applicazione di produzione distribuita in active Account AWS
- Conoscenza di base del formato di registrazione e dei modelli di eccezione dell'applicazione di produzione
- Log delle applicazioni configurati per lo streaming su Amazon CloudWatch Logs

Limitazioni

- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità regionale, vedi AWS Servizi per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.

Architecture

Il diagramma seguente mostra come CloudWatch Logs Insights valuta i log delle risorse e invia una visualizzazione dei dati pertinente a una dashboard. CloudWatch

Il diagramma mostra il flusso di lavoro seguente:

1. Le risorse pubblicano i log in Logs. CloudWatch Le risorse possono includere AWS risorse come istanze Amazon Elastic Compute Cloud (Amazon EC2) o bucket Amazon Simple Storage Service (Amazon S3). Un altro esempio include sistemi locali con CloudWatch Agent installato su cui è possibile pubblicare i log. CloudWatch
2. CloudWatch Filtri Logs Insights per la stringa di pattern pertinente. Esempi di stringhe di pattern di ricerca includono «error», «exception» o un'espressione regolare specifica.
3. In genere, il team di supporto alla produzione o gli sviluppatori aggiungono la visualizzazione del pattern alla CloudWatch dashboard.

Automazione e scalabilità

Gli sviluppatori possono automatizzare la soluzione di questo pattern utilizzando AWS Cloud Development Kit (AWS CDK) AWS CloudFormation, o AWS SDKs per gestire più modelli di stringhe.

I team possono incorporare questa automazione nei loro processi di integrazione e distribuzione continue (CI/CD). DevOps

Tools (Strumenti)

Servizi AWS

- Amazon CloudWatch Logs ti aiuta a centralizzare i log di tutti i tuoi sistemi e applicazioni, Servizi AWS così puoi monitorarli e archiviarli in modo sicuro.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.

Best practice

Efficienza delle query

- Definisci e configura i gruppi di log per analizzare i dati di registro pertinenti.
- Utilizza gli esploratori di campi per comprendere la struttura e i campi disponibili nei dati di registro.
- Scrivi query efficienti utilizzando la sintassi delle query di CloudWatch Logs Insights.
- Adatta le query di esempio alle tue esigenze specifiche per un'analisi più rapida.
- Limita gli intervalli di tempo di interrogazione per ridurre i dati scansionati e migliorare le prestazioni.
- Salva le interrogazioni per utilizzi futuri per risparmiare tempo e garantire un'analisi coerente.

Sicurezza

- Applica le politiche IAM appropriate a CloudWatch Logs Insights e ai gruppi di log. Segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.
- Abilita la crittografia dei dati di registro utilizzando AWS KMS i dati di registro sensibili.

Ottimizzazione dei costi

- CloudWatch Logs Insights addebita per GB di dati scansionati per query. Riduci gli intervalli di tempo e scegli come target gruppi di log specifici per ridurre i costi.
- Configura politiche di conservazione dei log appropriate per gestire i costi di storage.
- Per un'analisi frequente di set di dati storici di grandi dimensioni, prendi in considerazione l'esportazione dei log in Amazon S3 e l'utilizzo di Amazon Athena.
- Consulta CloudWatch i prezzi per comprendere le implicazioni in termini di costi per il tuo caso d'uso.

Epiche

Crea un gruppo di log e configura i log da visualizzare nella dashboard.

Operazione	Description	Competenze richieste
Configura le autorizzazioni IAM.	Per configurare le autorizza zioni IAM, procedi come segue: 1. Identifica i ruoli IAM utilizzat i dalla tua applicazione per scrivere i log e dagli utenti o dai servizi che creeranno dashboard, query e allarmi. 2. Per il ruolo dell'appl icazione, aggiungi le seguenti azioni e risorse API a una policy: `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [`	Amministratore AWS, AWS DevOps, amministratore di sistema AWS, amministr atore cloud, architetto cloud, DevOps ingegnere

Operazione	Description	Competenze richieste

```
 "logs:CreateLogGro
up",

 "logs:CreateLogStr
eam",

 "logs:PutLogEvents"
                ],

 "Resource": [

 "arn:aws:logs:*:*:
*"
                ]
            }
        ]
    }
```

3. Per la gestione CloudWatch delle risorse, aggiungi quanto segue a una policy:

```
{
    "Version":
 "2012-10-17",
    "Statement": [
        {
            "Effect":
 "Allow",
            "Action":
 [

 "cloudwatch:PutDas
hboard",

 "cloudwatch:GetDas
hboard",

 "logs:StartQuery",
```

Operazione	Description	Competenze richieste

	<pre>"logs:StopQuery",
	
	"logs:GetQueryResu
	lts",
	
	"cloudwatch:PutMet
	ricAlarm"
	],
	"Resource
	": "*"
	}
	]
	}</pre>
	Per informazioni su come creare policy IAM o aggiunger e autorizzazioni a policy esistenti, consulta Definizio ne delle autorizzazioni IAM personalizzate con policy gestite dai clienti e Modifica delle policy IAM nella IAM User Guide. Per ulteriori informazioni, consulta il riferimento alla gestione delle identità e degli accessi per Amazon CloudWatch Logs e alle autorizzazioni CloudWatch Logs nella Amazon CloudWatc h Logs User Guide.	

Operazione	Description	Competenze richieste

Creazione di un gruppo di log.	Per creare un gruppo di log, utilizza una delle seguenti opzioni:
	• Crea o apri un file CloudFormation modello (formato YAML o JSON) con estensione or. `.yaml` `.json` (Il codice seguente utilizza il formato YAML.) Aggiungi la seguente definizione di risorsa alla sezione Risorse del tuo modello:
	`MyLogGroup:`
	`  Type: AWS::Logs`
	`::LogGroup`
	`  Properties:`
	`  LogGroupName: my-`
	`log-group`
	Per ulteriori informazioni, consulta Quick Start: AWS CloudFormation Use to start with CloudWatch Logs nella Amazon CloudWatch Logs User Guide.
	• Usa la CloudWatch console, come descritto in Creare un gruppo di log in CloudWatch Logs nella Amazon CloudWatch Logs User Guide.
		Amministratore AWS, AWS DevOps, amministratore di sistema AWS, amministratore cloud, architetto cloud, DevOps ingegnere

Operazione	Description	Competenze richieste
	• Usa l'CreateLog Groupoperazione nell'API CloudWatch Logs.	

Operazione	Description	Competenze richieste
Genera una query CloudWatch Logs Insights.	Per creare e salvare una query di CloudWatch Logs Insights, procedi come segue: 1. Apri la CloudWatch console. 2. Vai a Logs Insights. 3. Crea la tua query utilizzando uno dei seguenti metodi: • Inizia con una query di esempio dalla sezione Interrogazioni. • Scrivi una query personalizzata. • Modifica la seguente query di esempio: `fields @timestamp, @message | filter @message like /(?i)exception/|stats count() by bin(30s)` Questa query controlla il file di registro, estrae il timestamp e il contenuto del messaggio, filtra per la parola «eccezione» (senza distinzione tra maiuscole e minuscole) e conta le occorrenze a intervalli di 30 secondi.	Amministratore AWS, AWS DevOps, amministratore di sistema AWS, amministratore cloud, architetto cloud, DevOps ingegnere

Operazione	Description	Competenze richieste
Crea visualizzazioni in una CloudWatch dashboard.	Per utilizzare una CloudWatch dashboard per creare una visualizzazione, procedi come segue: 1. Apri la CloudWatch console e crea una CloudWatch dashboard (se non ne hai già una) oppure apri una dashboard esistente. 2. Aggiungi la visualizzazione del monitoraggio delle eccezioni. Sono disponibili grafici a barre, a linee e a torta, a seconda delle esigenze aziendali. Per ulteriori informazioni, consulta Using widgets on CloudWatch dashboard nella Amazon CloudWatch Logs User Guide. Per la visualizzazione dei dati in tempo reale, puoi personalizzare un widget in base alle tue esigenze. Per ulteriori informazioni sulle opzioni e le funzionalità delle dashboard, consulta Using Amazon CloudWatch dashboard e Creazione di CloudWatch dashboard flessibili con variabili di	Amministratore AWS, AWS DevOps, amministratore di sistema AWS, amministratore cloud, architetto cloud, DevOps ingegnere

Operazione	Description	Competenze richieste
	dashboard nella Amazon CloudWatch Logs User Guide.	

risoluzione dei problemi

Problema	Soluzione
Impossibile visualizzare i risultati della query o la query sembra interrotta	Inizia con una query di lavoro che è stata modificata da una query di esempio. Eseguite piccole modifiche incrementali a parti della query (come un filtro o un campo) e sfruttate la funzionalità CloudWatch Logs Query Generator .
Gruppi di log che non creano flussi di log	Nella policy IAM, assicurati che la risorsa per le CreateLogStreamCreateLogGroupoperazioni sia fornita con un valore di caratteri (`*`) jolly. L'`create` operazione non avrà successo senza questa autorizzazione con i caratteri jolly.
Timeout delle query o rallentamento delle prestazioni	Riduci l'intervallo di tempo, scegli come target gruppi di log specifici o semplifica la query. I modelli complessi di espressioni regolari (`regex`) e gli ampi intervalli di tempo aumentano il tempo di interrogazione.
Nessun dato restituito per un intervallo di tempo valido	Verifica la selezione del gruppo di log e controlla che i log vengano inseriti (esamina i flussi di log) e conferma che lo schema di filtro corrisponda al formato di registro.

Risorse correlate

- Analisi dei dati di registro con Logs Insights CloudWatch
- Amazon CloudWatch FAQs
- Creazione di CloudWatch dashboard flessibili con variabili di dashboard
- Inizia a usare Logs Insights QL: Query tutorial
- Usa il linguaggio naturale per generare e aggiornare CloudWatch le query di Logs Insights
- Utilizzo PutDashboard con un AWS SDK o una CLI
- Lavorare con gruppi di log e flussi di log

Monitora l'uso di un'Amazon Machine Image condivisa su più Account AWS

Naveen Suthar e Sandeep Gawande, Amazon Web Services

Riepilogo

Amazon Machine Images (AMIs) vengono utilizzate per creare istanze Amazon Elastic Compute Cloud (Amazon EC2) nel tuo ambiente Amazon Web Services (AWS). Puoi creare AMIs un account AWS separato e centralizzato, chiamato account creator in questo modello. È quindi possibile condividere l'AMI tra più account Account AWS che si trovano nella stessa Regione AWS area, denominati account consumer secondo questo schema. La gestione AMIs da un unico account offre scalabilità e semplifica la governance. Negli account consumer, puoi fare riferimento all'AMI condivisa nei modelli di lancio di Amazon EC2 Auto Scaling e nei gruppi di nodi Amazon Elastic Kubernetes Service (Amazon EKS).

Quando un'AMI condivisa è obsoleta, cancellata o non è condivisa, quelle che Servizi AWS fanno riferimento all'AMI negli account consumer non possono utilizzare questa AMI per avviare nuove istanze. Qualsiasi evento di ridimensionamento automatico o riavvio della stessa istanza ha esito negativo. Ciò può causare problemi nell'ambiente di produzione, come tempi di inattività delle applicazioni o peggioramento delle prestazioni. Quando gli eventi di condivisione e utilizzo dell'AMI si verificano più volte Account AWS, può essere difficile monitorare questa attività.

Questo modello consente di monitorare l'utilizzo e lo stato delle AMI condivise tra gli account nella stessa regione. Utilizza sistemi serverless Servizi AWS, come Amazon EventBridge, Amazon AWS Lambda DynamoDB e Amazon Simple Email Service (Amazon SES). Effettua il provisioning dell'infrastruttura come codice (IaC) utilizzando Terraform. HashiCorp Questa soluzione fornisce avvisi quando un servizio in un account consumatore fa riferimento a un'AMI non registrata o non condivisa.

Prerequisiti e limitazioni

Prerequisiti

- Due o più account attivi Account AWS: un account Creator e uno o più account consumer
- Uno o più AMIs condivisi dall'account creatore a un account consumatore
- Terraform CLI, installata (documentazione Terraform)
- Terraform AWS Provider, configurato (documentazione Terraform)
- (Facoltativo, ma consigliato) Backend Terraform, configurato (documentazione Terraform)

- Git, installato

Limitazioni

- Questo modello monitora AMIs ciò che è stato condiviso con account specifici utilizzando l'ID dell'account. Questo modello non monitora AMIs ciò che è stato condiviso con un'organizzazione utilizzando l'ID dell'organizzazione.
- AMIs può essere condiviso solo con account che si trovano all'interno della stessa Regione AWS. Questo modello esegue il monitoraggio AMIs all'interno di un'unica regione di destinazione. Per monitorare l'utilizzo AMIs in più regioni, è necessario implementare questa soluzione in ciascuna regione.
- Questo modello non monitora nulla di AMIs ciò che è stato condiviso prima dell'implementazione di questa soluzione. Se desideri monitorare i dati precedentemente condivisi AMIs, puoi annullare la condivisione dell'AMI e ricondividerla con gli account utente.

Versioni del prodotto

- Terraform versione 1.2.0 o successiva
- Terraform AWS Provider versione 4.20 o successiva

Architecture

Stack tecnologico Target

Le seguenti risorse vengono fornite come IaC tramite Terraform:

- Tabelle Amazon DynamoDB
- EventBridge Regole di Amazon
- AWS Identity and Access Management ruolo (IAM)
- AWS Lambda funzioni
- Amazon SES

Architettura di destinazione

Il diagramma mostra il flusso di lavoro seguente:

1. Un'AMI nell'account del creatore viene condivisa con un account consumatore nello stesso Regione AWS.
2. Quando l'AMI è condivisa, una EventBridge regola nell'account creatore acquisisce l'`ModifyImageAttribute`evento e avvia una funzione Lambda nell'account creatore.
3. La funzione Lambda archivia i dati relativi all'AMI in una tabella DynamoDB nell'account creatore.
4. Quando un Servizio AWS account consumer utilizza l'AMI condivisa per avviare un' EC2 istanza Amazon o quando l'AMI condivisa è associata a un modello di avvio, una EventBridge regola nell'account consumer rileva l'uso dell'AMI condivisa.
5. La EventBridge regola avvia una funzione Lambda nell'account consumer. La funzione Lambda; svolge le operazioni seguenti:
   a. La funzione Lambda aggiorna i dati relativi all'AMI in una tabella DynamoDB nell'account consumer.
   b. La funzione Lambda assume un ruolo IAM nell'account creatore e aggiorna la tabella Lambda nell'account creatore. Nella `Mapping` tabella, crea un elemento che associa l'ID dell'istanza o l'ID del modello di avvio al rispettivo ID AMI.
6. L'AMI gestita centralmente nell'account del creatore è obsoleta, cancellata o non è condivisa.
7. La EventBridge regola nell'account creatore acquisisce l'`DeregisterImage`evento `ModifyImageAttribute` or con l'`remove`azione e avvia la funzione Lambda.
8. La funzione Lambda controlla la tabella DynamoDB per determinare se l'AMI viene utilizzata in uno qualsiasi degli account consumer. Se nella `Mapping` tabella non è presente alcuna istanza IDs o modello di avvio IDs associato all'AMI, il processo è completo.
9. Se un'istanza IDs o un modello di avvio IDs sono associati all'AMI nella `Mapping` tabella, la funzione Lambda utilizza Amazon SES per inviare una notifica e-mail agli abbonati configurati.

Tools (Strumenti)

Servizi AWS

- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altro modo. Account AWS

- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Simple Email Service (Amazon SES) Simple Email Service (Amazon SES) ti aiuta a inviare e ricevere e-mail utilizzando i tuoi indirizzi e-mail e domini.

Altri strumenti

- HashiCorp Terraform è uno strumento Infrastructure as Code (IaC) che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud.
- Python è un linguaggio di programmazione per computer generico.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub cross-account-ami-monitoring-terraform-samples.

Best practice

- Segui le migliori pratiche per lavorare con le funzioni. AWS Lambda
- Segui le migliori pratiche per la costruzione AMIs.
- Quando crei il ruolo IAM, segui il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le migliori pratiche relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.
- Imposta il monitoraggio e gli avvisi per le AWS Lambda funzioni. Per ulteriori informazioni, consulta Monitoraggio e risoluzione dei problemi delle funzioni Lambda.

Epiche

Personalizza i file di configurazione Terraform

Operazione	Description	Competenze richieste
Crea i profili AWS CLI denominati.	Per l'account creatore e ogni account consumatore, crea un profilo denominato AWS Command Line Interface (AWS CLI). Per istruzioni, consulta Configurare il AWS CLI nel Centro risorse AWS introduttive.	DevOps ingegnere
Clonare il repository.	Inserire il seguente comando. Questo clona il repository cross-account-ami-monitoring-terraform-samples utilizzando SSH. GitHub `git clone git@github.com:aws-samples/cross-account-ami-monitoring-terraform-samples.git`	DevOps ingegnere
Aggiorna il file provider.tf.	1. Immettete il seguente comando per navigare nella `terraform` cartella del repository clonato. `cd cross-account-ami-monitoring/terraform` 2. Apri il file `provider.tf` . 3. Aggiorna le configurazioni Terraform AWS Provider	DevOps ingegnere

Operazione	Description	Competenze richieste
	per l'account creatore e l'account consumatore come segue: • Per `alias`, inserisci un nome per la configura zione del provider. • Per `region`, inserisci la destinazione Regione AWS in cui desideri implementare questa soluzione. • Per `profile`, inserisci il profilo AWS CLI denominato per accedere all'account. 4. Se stai configurando più di un account consumato re, crea un profilo per ogni account consumatore aggiuntivo. 5. Salvare e chiudere il file `provider.tf` . Per ulteriori informazioni sulla configurazione dei provider, consulta Configurazioni di più provider nella documentazione di Terraform.	

Operazione	Description	Competenze richieste
Aggiorna il file terraform.tfvars.	1. Apri il file `terraform.tfvars` . 2. Nel `account_email_mapping` parametro, configura gli avvisi per l'account creatore e l'account consumatore come segue: • Per `account`, inserisci l'ID dell'account. • Per `email`, inserisci l'indirizzo email a cui desideri inviare gli avvisi. Puoi inserire un solo indirizzo email per ogni account. 3. Se stai configurando più di un account consumatore, inserisci un account e un indirizzo e-mail per ogni account consumatore aggiuntivo. 4. Salvare e chiudere il file `terraform.tfvars` .	DevOps ingegnere

Operazione	Description	Competenze richieste
Aggiorna il file.tf principale.	Completa questi passaggi solo se stai distribuendo questa soluzione su più di un account consumatore. Se si implement a questa soluzione su un solo account consumer, non è necessaria alcuna modifica di questo file. 1. Apri il file `main.tf`. 2. Per ogni account consumatore aggiuntiv o, crea un nuovo modulo basato sul `consumer_ account_A` modulo del modello. Per ogni account consumatore, for`provider`, il valore deve corrispondere all'alias inserito nel `provider.tf` file. 3. Salvare e chiudere il file `main.tf`.	DevOps ingegnere

Implementa la soluzione utilizzando Terraform

Operazione	Description	Competenze richieste
Distribuire la soluzione.	Nella CLI Terraform, inserisci i seguenti comandi per distribui re le risorse AWS negli account creatore e consumato re:	DevOps ingegnere

Operazione	Description	Competenze richieste

	1. Inserisci il seguente comando per inizializzare Terraform.
	`terraform init`
	2. Immettere il seguente comando per convalidare le configurazioni Terraform.
	`terraform validate`
	3. Immettere il seguente comando per creare un piano di esecuzione Terraform.
	`terraform plan`
	4. Rivedi le modifiche alla configurazione nel piano Terraform e conferma che desideri implementare queste modifiche.
	5. Immettere il seguente comando per distribuire le risorse.
	`terraform apply`	

Operazione	Description	Competenze richieste
Verifica l'identità dell'indirizzo e-mail.	Quando hai implementato il piano Terraform, Terraform ha creato un indirizzo e-mail per ogni account consumer in Amazon SES. Prima di poter inviare notifiche a quell'ind irizzo e-mail, devi verificare l'indirizzo e-mail. Per istruzion i, consulta Verifica dell'iden tità di un indirizzo e-mail nella documentazione di Amazon SES.	Informazioni generali su AWS

Convalida la distribuzione delle risorse

Operazione	Description	Competenze richieste
Convalida la distribuzione nell'account creatore.	1. Accedi all'account creatore. 2. Nella barra di navigazio ne, conferma che stai visualizzando l'obiettivo Regione AWS. Se ti trovi in una regione diversa, scegli il nome della regione attualmente visualizzata, quindi scegli la regione di destinazione. 3. Aprire la console DynamoDB. 4. Nel pannello di navigazione, seleziona Tabelle.	DevOps ingegnere

Operazione	Description	Competenze richieste
	5. Nell'elenco delle tabelle, verifica che la `AmiShare` tabella sia presente. 6. Aprire la console Lambda. 7. Nel riquadro di navigazione, seleziona Funzioni. 8. Nell'elenco delle funzioni, verifica che la `ami-share` funzione sia presente. 9. Aprire la console IAM. 10. Nel pannello di navigazione, selezionare Ruoli. 11. Nell'elenco dei ruoli, verifica che il `external-ddb-role` ruolo sia presente. 12. Apri la EventBridge console. 13. Nel riquadro di navigazione, scegli Regole. 14. Nell'elenco delle regole, verifica che la `modify_image_attribute_event` regola sia presente. 15. Apri la console Amazon SES. 16. Nel riquadro di navigazione, scegli Identità verificate. 17. Nell'elenco delle identità, verifica che l'identità di un indirizzo e-mail sia stata registrata e verificata per ogni account consumatore.	

Operazione	Description	Competenze richieste
Convalida l'implementazione nell'account consumer.	1. Accedi all'account consumer. 2. Nella barra di navigazione, conferma che stai visualizzando la destinazione Regione AWS. Se ti trovi in una regione diversa, scegli il nome della regione attualmente visualizzata, quindi scegli la regione di destinazione. 3. Aprire la console DynamoDB. 4. Nel pannello di navigazione, seleziona Tabelle. 5. Nell'elenco delle tabelle, verifica che la `Mapping` tabella sia presente. 6. Aprire la console Lambda. 7. Nel riquadro di navigazione, seleziona Funzioni. 8. Nell'elenco delle funzioni, verifica che siano presenti `ami-deregister-function` le funzioni `ami-usage-function` and. 9. Apri la EventBridge console. 10. Nel riquadro di navigazione, scegli Regole. 11. Nell'elenco delle regole, verifica che siano presenti	DevOps ingegnere

Operazione	Description	Competenze richieste
	`ami_deregister_events` le regole `ami_usage_events` and.	

Convalida il monitoraggio

Operazione	Description	Competenze richieste
Crea un'AMI nell'account del creatore.	1. Nell'account del creatore, crea un'AMI privata. Per istruzioni, consulta Creare un'AMI da un' EC2 istanza Amazon. 2. Condividi la nuova AMI con uno degli account consumer. Per istruzioni, consulta Condividere un'AMI con specifiche Account AWS.	DevOps ingegnere
Usa l'AMI nell'account del consumatore.	Nell'account consumer, utilizza l'AMI condivisa per creare un' EC2 istanza Amazon o un modello di avvio. Per istruzioni, consulta Come posso avviare un' EC2 istanza Amazon da un'AMI personalizzata (AWS re:POST Knowledge Center) o Creare un modello di avvio per un gruppo di Auto Scaling (documentazione di Amazon Auto EC2 Scaling).	DevOps ingegnere

Operazione	Description	Competenze richieste
Convalida il monitoraggio e gli avvisi.	1. Accedi all'account del creatore. 2. Apri la EC2 console Amazon. 3. Nel pannello di navigazione, scegli AMIs. 4. Seleziona l'AMI nell'elenco, quindi scegli Azioni, Modifica autorizzazioni AMI. 5. Nella sezione Account condivisi, seleziona l'account consumatore, quindi scegli Rimuovi selezionato. 6. Scegli Save changes (Salva modifiche). 7. Verifica che l'indirizzo e-mail di destinazione che hai definito per l'account consumer riceva una notifica che la condivisione è stata annullata per l'AMI.	DevOps ingegnere

(Facoltativo) Interrompi il monitoraggio condiviso AMIs

Operazione	Description	Competenze richieste
Eliminare le risorse.	1. Immettere il seguente comando per rimuovere le risorse distribuite secondo questo schema e interrompere il monitoraggio condiviso AMIs.	DevOps ingegnere

Operazione	Description	Competenze richieste
	`terraform destroy` 2. Conferma il `destroy` comando `yes` inserendo.	

risoluzione dei problemi

Problema	Soluzione
Non ho ricevuto un avviso via e-mail.	Potrebbero esserci diversi motivi per cui l'e-mail di Amazon SES non è stata inviata. Verifica quanto segue: 1. Nella sezione Epics, usa l'epic Validate resource deployment per confermare che l'infrastruttura sia stata completamente fornita correttamente. Account AWS 2. Convalida gli eventi della funzione Lambda in Amazon CloudWatch Logs. Per istruzioni, consulta Uso della CloudWatch console nella documentazione di Lambda. Verifica che non vi siano problemi di autorizzazione, ad esempio una negazione esplicita in qualsiasi politica basata sull'identità o basata sulle risorse. Per ulteriori informazioni, consulta Logica di valutazione delle politiche nella documentazione IAM. 3. In Amazon SES, verifica che lo stato dell'identità dell'indirizzo e-mail sia Verificat o. Per ulteriori informazioni, consulta Verifica dell'identità di un indirizzo e-mail.

Risorse correlate

AWS documentazione

- Creazione di funzioni Lambda con Python (documentazione Lambda)
- Creare un AMI (EC2 documentazione Amazon)
- Condividi un'AMI con specifiche Account AWS (EC2 documentazione Amazon)
- Annulla la registrazione dell'AMI (documentazione Amazon EC2)

Documentazione Terraform

- Installa Terraform
- Configurazione del backend Terraform
- Fornitore AWS Terraform
- Scarica il file binario di Terraform

Visualizza i dettagli degli snapshot EBS per il tuo account o la tua organizzazione AWS

Arun Chandapillai e Parag Nagwekar, Amazon Web Services

Riepilogo

Questo modello descrive come generare automaticamente un report su richiesta di tutte le snapshot di Amazon Elastic Block Store (Amazon EBS) nel tuo account Amazon Web Services (AWS) o unità organizzativa (OU) in AWS Organizations.

Amazon EBS è un easy-to-use servizio di storage a blocchi scalabile e ad alte prestazioni progettato per Amazon Elastic Compute Cloud (Amazon). EC2 Un volume EBS fornisce uno storage durevole e persistente che puoi collegare alle tue istanze. EC2 Puoi utilizzare i volumi EBS come storage principale per i tuoi dati ed eseguire un point-in-time backup dei tuoi volumi EBS creando un'istantanea. Puoi utilizzare la Console di gestione AWS o l'AWS Command Line Interface (AWS CLI) per visualizzare i dettagli di snapshot EBS specifici. Questo modello fornisce un modo programmatico per recuperare informazioni su tutte le snapshot EBS nel tuo account AWS o nell'unità organizzativa.

Puoi utilizzare lo script fornito da questo pattern per generare un file con valori separati da virgole (CSV) contenente le seguenti informazioni su ogni snapshot: ID account, ID snapshot, ID e dimensione del volume, data di acquisizione dello snapshot, ID dell'istanza e descrizione. Se le tue istantanee EBS sono contrassegnate, il rapporto include anche gli attributi del proprietario e del team.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- AWS CLI versione 2 installata e configurata
- Ruolo AWS Identity and Access Management (IAM) con le autorizzazioni appropriate (autorizzazioni di accesso per un account specifico o per tutti gli account in un'unità organizzativa se prevedi di eseguire lo script da AWS Organizations)

Architecture

Il diagramma seguente mostra il flusso di lavoro dello script che genera un report su richiesta di snapshot EBS distribuiti su più account AWS in un'unità organizzativa.

Tools (Strumenti)

Servizi AWS

- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.
- Amazon Elastic Block Store (Amazon EBS) fornisce volumi di storage a livello di blocco da utilizzare con le istanze. EC2
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Organizations è un servizio di gestione degli account che ti aiuta a consolidare più account AWS in un'organizzazione da creare e gestire centralmente.

Codice

Il codice per l'applicazione di esempio utilizzata in questo modello è disponibile su GitHub, nel aws-ebs-snapshots-awsorganizationsrepository. Segui le istruzioni nella sezione successiva per utilizzare i file di esempio.

Epiche

Scarica lo script

Operazione	Description	Competenze richieste
Scarica lo script Python.	Scarica lo script GetSnapshotDetailsAllAccountsOU.py dal GitHub repository.	Informazioni generali su AWS

Ottieni i dettagli dello snapshot EBS per un account AWS

Operazione	Description	Competenze richieste
Eseguire lo script Python.	Eseguire il comando :	Informazioni generali su AWS

Operazione	Description	Competenze richieste

	`python3 getsnapsh otinfo.py --file <output-file>.csv --region <region-name>`
	
	dove `<output-file>` si riferisce al file di output CSV in cui desideri inserire le informazioni sugli snapshot EBS ed è `<region-name>` la regione AWS in cui sono archiviate le istantanee. Esempio:
	
	`python3 getsnapsh otinfo.py --file snapshots.csv --region us-east-1`	

Ottieni i dettagli delle istantanee EBS per un'organizzazione

Operazione	Description	Competenze richieste
Eseguire lo script Python.	Eseguire il comando : `python3 getsnapsh otinfo.py --file <output-file>.csv --role <IAM-role> --region <region-name>` dove `<output-file>` si riferisce al file di output CSV in cui desideri inserire le informazioni sugli snapshot	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	EBS, `<IAM-role>` è un ruolo che fornisce le autorizzazioni per accedere ad AWS Organizations ed è `<region-name>` la regione AWS in cui sono archiviate le istantanee. Esempio: `python3 getsnapshotinfo.py --file snapshots.csv --role <IAM role> --region us-west-2`	

Risorse correlate

- Documentazione Amazon EBS
- Azioni Amazon EBS
- Riferimento all'API Amazon EBS
- Miglioramento delle prestazioni di Amazon EBS
- Risorse Amazon EBS
- Prezzi degli snapshot EBS

Informazioni aggiuntive

Tipi di istantanee EBS

Amazon EBS offre tre tipi di snapshot, in base alla proprietà e all'accesso:

- Di tua proprietà: per impostazione predefinita, solo tu puoi creare volumi da istantanee di tua proprietà.
- Istantanee pubbliche: puoi condividere istantanee pubblicamente con tutti gli altri account AWS. Per creare uno snapshot pubblico, modifichi le autorizzazioni per uno snapshot per condividerlo con gli account AWS da te specificati. Gli utenti che autorizzerai possono quindi utilizzare gli

snapshot che condividi creando i propri volumi EBS, mentre lo snapshot originale rimane inalterato. Puoi anche rendere le tue istantanee non crittografate disponibili pubblicamente a tutti gli utenti AWS. Tuttavia, non puoi rendere le tue istantanee crittografate disponibili pubblicamente per motivi di sicurezza. Le istantanee pubbliche rappresentano un rischio significativo per la sicurezza a causa della possibilità di esporre dati personali e sensibili. Ti consigliamo vivamente di non condividere le tue istantanee EBS con tutti gli account AWS. Per ulteriori informazioni sulla condivisione di snapshot, consulta la documentazione AWS.

- Istantanee private: puoi condividere istantanee in privato con singoli account AWS da te specificati. Per condividere lo snapshot in privato con account AWS specifici, segui le istruzioni nella documentazione AWS e scegli Privato per l'impostazione delle autorizzazioni. Gli utenti autorizzati possono utilizzare lo snapshot condiviso per la creazione di propri volumi EBS, mentre lo snapshot originale rimane inalterato.

Panoramiche e procedure

La tabella seguente fornisce collegamenti a ulteriori informazioni sulle istantanee EBS, incluso come ridurre i costi in termini di volume EBS trovando ed eliminando le istantanee non utilizzate e archiviare le istantanee a cui si accede raramente che non richiedono un recupero frequente o rapido.

Per informazioni su	See
Istantanee, relative funzionalità e limitazioni	Crea istantanee Amazon EBS
Come creare uno snapshot	Console: crea un'istantanea CLI AWS: comando create-snapshot Esempio: `aws ec2 create-snapshot --volume-id vol-1234567890abcdef0 --description " volume snapshot"`
Eliminazione di istantanee (informazioni generali)	Eliminare uno snapshot di Amazon EBS
Come eliminare uno snapshot	Console: elimina un'istantanea

	CLI AWS: comando delete-snapshot Esempio: `aws ec2 delete-snapshot --snapshot-id snap-1234567890abcdef0`
Archiviazione delle istantanee (informazioni generali)	Archivia gli snapshot di Amazon EBS Amazon EBS Snapshots Archive (post di blog)
Come archiviare uno snapshot	Console: archivia un'istantanea AWS CLI: comando modify-snapshot-tier
Come recuperare un'istantanea archiviata	Console: ripristina un'istantanea archiviata AWS CLI: comando restore-snapshot-tier
Prezzi delle istantanee	Prezzi di Amazon EBS

DOMANDE FREQUENTI

Qual è il periodo minimo di archiviazione?

Il periodo minimo di archiviazione è di 90 giorni.

Quanto tempo occorre per ripristinare un'istantanea archiviata?

Possono essere necessarie fino a 72 ore per ripristinare uno snapshot archiviato dal livello archivio al livello standard, a seconda delle dimensioni dello snapshot.

Le istantanee archiviate sono istantanee complete?

Gli snapshot archiviati sono sempre snapshot completi.

Quali istantanee può archiviare un utente?

È possono archiviare solo gli snapshot che possiedi nel proprio account.

È possibile archiviare un'istantanea del volume del dispositivo root di un'Amazon Machine Image (AMI) registrata?

No, non è possibile archiviare un'istantanea del volume del dispositivo principale di un'AMI registrata.

Quali sono le considerazioni sulla sicurezza per la condivisione di un'istantanea?

Quando condividi un'istantanea, consenti ad altri di accedere a tutti i dati dell'istantanea. Condividi le istantanee solo con persone di cui ti fidi per i tuoi dati.

Come si condivide uno snapshot con un'altra regione AWS?

Gli snapshot sono vincolati alla regione in cui sono stati creati. Per condividere uno snapshot con altre Regioni, copia lo snapshot nella regione desiderata.

È possibile condividere istantanee crittografate?

Non puoi condividere istantanee crittografate con la chiave gestita AWS predefinita. Puoi condividere istantanee crittografate solo con una chiave gestita dal cliente. Quando si condivide un'istantanea crittografata, è necessario condividere anche la chiave gestita dal cliente utilizzata per crittografare l'istantanea.

Che dire delle istantanee non crittografate?

È possibile condividere pubblicamente istantanee non crittografate.

Altri modelli

- Automatizza la creazione di account utilizzando Landing Zone Accelerator su AWS
- Automatizza le operazioni AWS dell'infrastruttura utilizzando Amazon Bedrock
- Automatizza la valutazione delle risorse AWS
- Inventaria automaticamente AWS le risorse su più account e regioni
- Automatizza il portafoglio e la distribuzione dei prodotti di AWS Service Catalog utilizzando AWS CDK
- Automatizza la gestione dinamica delle pipeline per l'implementazione di soluzioni hotfix in ambienti Gitflow utilizzando e AWS Service Catalog AWS CodePipeline
- Automatizza l'inserimento e la visualizzazione di metriche personalizzate di Amazon MWAA su Amazon Managed Grafana utilizzando Terraform
- Associa automaticamente una policy gestita da AWS per Systems Manager ai profili di EC2 istanza utilizzando Cloud Custodian e AWS CDK
- Crittografa automaticamente i volumi Amazon EBS esistenti e nuovi
- Crea una AWS landing zone che includa MongoDB Atlas
- Centralizza il monitoraggio utilizzando Amazon CloudWatch Observability Access Manager
- Verifica la presenza di tag obbligatori EC2 nelle istanze al momento del lancio
- Pulisci le risorse di AWS Account Factory for Terraform (AFT) in modo sicuro dopo la perdita dei file di stato
- Crea una definizione di attività Amazon ECS e monta un file system su EC2 istanze utilizzando Amazon EFS
- Crea regole AWS Config personalizzate utilizzando AWS CloudFormation Guard le politiche
- Personalizza i nomi dei ruoli predefiniti utilizzando AWS CDK gli aspetti e gli escape hatch
- Implementa e gestisci AWS Control Tower i controlli utilizzando AWS CDK e CloudFormation
- Implementa e gestisci AWS Control Tower i controlli utilizzando Terraform
- Distribuisci codice in più regioni AWS utilizzando AWS CodePipeline CodeCommit, AWS e AWS CodeBuild
- Distribuisci applicazioni containerizzate in AWS IoT Greengrass V2 esecuzione come contenitore Docker
- Abilita Amazon in GuardDuty modo condizionale utilizzando modelli AWS CloudFormation
- Abilita l'archiviazione dei DB2 log direttamente su Amazon S3 in un database IBM Db2

- Esporta un report delle identità di AWS IAM Identity Center e delle relative assegnazioni utilizzando PowerShell
- Genera un CloudFormation modello AWS contenente le regole gestite di AWS Config utilizzando Troposphere
- Offri alle istanze di SageMaker notebook l'accesso temporaneo a un CodeCommit repository in un altro account AWS
- Integra il controller universale Stonebranch con la modernizzazione del mainframe AWS
- Avvia un CodeBuild progetto su più account AWS utilizzando Step Functions e una funzione proxy Lambda
- Gestisci i set di AWS autorizzazioni in modo dinamico utilizzando Terraform
- Esegui la migrazione di applicazioni ospitate da IIS su Amazon utilizzando appcmd.exe EC2
- Migrazione dei certificati SSL di Windows su un Application Load Balancer utilizzando ACM
- Monitoraggio dell'attività dell'utente root IAM
- Crea un'architettura IPAM gerarchica e multiregionale utilizzando Terraform AWS
- Ottimizza le implementazioni serverless multi-account utilizzando i flussi di lavoro e Actions AWS CDK GitHub
- Conserva lo spazio IP instradabile nei progetti VPC multi-account per sottoreti non destinate ai carichi di lavoro
- Fornisci ruoli IAM con privilegi minimi implementando una soluzione di distribuzione automatica di ruoli
- Registrazione di più indirizzi e-mail Account AWS con un unico indirizzo e-mail utilizzando Amazon SES
- Rimuovi Amazon EC2 Entries Account AWS dall'altra AWS Managed Microsoft AD parte utilizzando l' AWS Lambda automazione
- Rimuovi EC2 le voci Amazon nello stesso Account AWS modulo AWS Managed Microsoft AD utilizzando l' AWS Lambda automazione
- Esegui carichi di lavoro pianificati e basati su eventi su larga scala con AWS Fargate
- Proteggi i dati sensibili nei CloudWatch registri utilizzando Amazon Macie
- Invia notifiche per un'istanza di database Amazon RDS for SQL Server utilizzando un server SMTP locale e Database Mail
- Configura una dashboard di monitoraggio Grafana per AWS ParallelCluster
- Configura la registrazione centralizzata su scala aziendale utilizzando Terraform

- Configura il disaster recovery per SAP su IBM Db2 su AWS
- Semplifica la gestione EC2 della conformità di Amazon con agenti Amazon Bedrock e AWS Config
- Etichetta automaticamente gli allegati Transit Gateway utilizzando AWS Organizations
- Usa le query BMC Discovery per estrarre i dati di migrazione per la pianificazione della migrazione
- Verifica le best practice operative per PCI DSS 4.0 utilizzando AWS Config
- Visualizza i log e i parametri di AWS Network Firewall utilizzando Splunk
- Visualizza i report sulle credenziali IAM per tutti gli account AWS utilizzando Amazon Quick Sight

Messaggi e comunicazioni

Argomenti

- Automatizza la configurazione di RabbitMQ in Amazon MQ
- Migliora la qualità delle chiamate sulle postazioni di lavoro degli agenti nei contact center Amazon Connect
- Altri modelli

Automatizza la configurazione di RabbitMQ in Amazon MQ

Yogesh Bhatia e Afroz Khan, Amazon Web Services

Riepilogo

Amazon MQ è un servizio di broker di messaggi gestito che offre compatibilità con molti broker di messaggi popolari. L'uso di Amazon MQ con RabbitMQ fornisce un robusto cluster RabbitMQ gestito nel cloud Amazon Web Services (AWS) con più broker e opzioni di configurazione. Amazon MQ fornisce un'infrastruttura altamente disponibile, sicura e scalabile e può elaborare un gran numero di messaggi al secondo con facilità. Più applicazioni possono utilizzare l'infrastruttura con diversi host virtuali, code e scambi. Tuttavia, la gestione di queste opzioni di configurazione o la creazione manuale dell'infrastruttura possono richiedere tempo e impegno. Questo modello descrive un modo per gestire le configurazioni per RabbitMQ in un unico passaggio, tramite un singolo file. È possibile incorporare il codice fornito con questo pattern in qualsiasi strumento di integrazione continua (CI) come Jenkins o Bamboo.

È possibile utilizzare questo modello per configurare qualsiasi cluster RabbitMQ. Tutto ciò che serve è la connettività al cluster. Sebbene esistano molti altri modi per gestire le configurazioni di RabbitMQ, questa soluzione crea intere configurazioni di applicazioni in un unico passaggio, in modo da poter gestire facilmente code e altri dettagli.

Prerequisiti e limitazioni

Prerequisiti

- AWS Command Line Interface (AWS CLI) installata e configurata in modo che punti al tuo account AWS (per istruzioni, consulta la documentazione AWS CLI)
- Ansible è installato, quindi puoi eseguire i playbook per creare la configurazione
- rabbitmqadmin installato (per istruzioni, consultate la documentazione di RabbitMQ)
- Un cluster RabbitMQ in Amazon MQ, creato con parametri Amazon affidabili CloudWatch

Requisiti aggiuntivi

- Assicurati di creare le configurazioni per gli host e gli utenti virtuali separatamente e non come parte di JSON.
- Assicurati che la configurazione JSON faccia parte del repository e sia controllata dalla versione.

- La versione della CLI di rabbitmqadmin deve essere la stessa del server RabbitMQ, quindi l'opzione migliore è scaricare la CLI dalla console RabbitMQ.
- Come parte della pipeline, assicurati che la sintassi JSON sia convalidata prima di ogni esecuzione.

Versioni del prodotto

- AWS CLI versione 2.0
- Ansible versione 2.9.13
- rabbitmqadmin versione 3.9.13 (deve essere la stessa della versione del server RabbitMQ)

Architecture

Stack tecnologico di origine

- Un cluster RabbitMQ in esecuzione su una macchina virtuale (VM) locale esistente o su un cluster Kubernetes (in locale o nel cloud)

Stack tecnologico Target

- Configurazioni RabbitMQ automatizzate su Amazon MQ per RabbitMQ

Architettura Target

Esistono molti modi per configurare RabbitMQ. Questo modello utilizza la funzionalità di importazione di configurazione, in cui un singolo file JSON contiene tutte le configurazioni. Questo file applica tutte le impostazioni e può essere gestito da un sistema di controllo delle versioni come Bitbucket o Git. Questo modello utilizza Ansible per implementare la configurazione tramite la CLI rabbitmqadmin.

Tools (Strumenti)

Strumenti

- rabbitmqadmin è uno strumento a riga di comando per l'API basata su HTTP di RabbitMQ. Viene utilizzato per gestire e monitorare i nodi e i cluster RabbitMQ.
- Ansible è uno strumento open source per l'automazione delle applicazioni e dell'infrastruttura IT.

- L'interfaccia a riga di comando di AWS consente di interagire con i servizi AWS utilizzando i comandi in una shell a riga di comando.

Servizi AWS

- Amazon MQ è un servizio di broker di messaggi gestito che semplifica la configurazione e la gestione di broker di messaggi nel cloud.
- AWS ti CloudFormation aiuta a configurare la tua infrastruttura AWS e ad accelerare il provisioning del cloud con l'infrastruttura come codice.

Codice

Il file di configurazione JSON utilizzato in questo modello e un esempio di playbook Ansible sono forniti in allegato.

Epiche

Crea la tua infrastruttura AWS

Operazione	Description	Competenze richieste
Crea un cluster RabbitMQ su AWS.	Se non disponi già di un cluster RabbitMQ, puoi utilizzare AWS CloudFormation per creare lo stack su AWS. In alternativa, puoi utilizzare il modulo Cloudformation in Ansible per creare lo stack. Con quest'ultimo approccio, puoi utilizzare Ansible per entrambe le attività: creare l'infrastruttura RabbitMQ e gestire le configurazioni.	AWS CloudFormation, Ansible

Creare la configurazione Amazon MQ per RabbitMQ

Operazione	Description	Competenze richieste
Creare un file delle proprietà.	Scarica il file di configurazione JSON (`rabbitmqconfig.json`) nell'allegato o esportalo dalla console RabbitMQ. Modificalo per configurare code, scambi e associazioni. Questo file di configurazione dimostra quanto segue: - Crea due code: e `sample-queue1` `sample-queue2` - Crea due scambi: `sample-exchange1` e `sample-exchange2` - Implementa l'associazione tra le code e gli scambi Queste configurazioni vengono eseguite sull'host virtuale root (/), come richiesto da rabbitmqadmin.	JSON
Recupera i dettagli dell'infrastruttura Amazon MQ for RabbitMQ.	Recupera i seguenti dettagli per l'infrastruttura RabbitMQ su AWS: • Nome broker • Host RabbitMQ • nome utente RabbitMQ (l'utente amministratore	CLI AWS, Amazon MQ

Operazione	Description	Competenze richieste
	creato durante la creazione del cluster) • Password RabbitMQ Puoi utilizzare la Console di gestione AWS o la CLI AWS per recuperare queste informazioni. Questi dettagli consentono al playbook Ansible di connettersi al tuo account AWS e utilizzare il cluster RabbitMQ per eseguire comandi. **Important** Il computer che esegue il playbook Ansible deve essere in grado di accedere al tuo account AWS e la CLI AWS deve essere già configurata, come descritto nella sezione Prerequisiti.	

Operazione	Description	Competenze richieste

Crea il file hosts_var.	Crea il `hosts_var` file per Ansible e assicurati che tutte le variabili siano definite nel file. Prendi in considerazione l'utilizzo di Ansible Vault per memorizzare la password. Puoi configurare il `hosts_var` file come segue (sostitui sci gli asterischi con le tue informazioni):
	`RABBITMQ_HOST: "**********.mq.us-east-2.amazonaws.com"`
	`RABBITMQ_VHOST: "/"`
	`RABBITMQ_USERNAME: "admin"`
	`RABBITMQ_PASSWORD: "*******"`
		Ansible

Operazione	Description	Competenze richieste
Crea un playbook Ansible.	Per un playbook di esempio, vedi `ansible-rabbit-config.yaml` in allegato. Scarica e salva questo file. Il playbook Ansible importa e gestisce tutte le configurazioni di RabbitMQ, come code, scambi e associazioni, richieste dalle applicazioni. Segui le migliori pratiche per i playbook Ansible, come la protezione delle password. Usa Ansible Vault per la crittografia delle password e recupera la password di RabbitMQ dal file crittografato.	Ansible

Distribuzione della configurazione

Operazione	Description	Competenze richieste
Esegui il playbook.	Esegui il playbook Ansible che hai creato nell'epopea precedente. `ansible-playbook ansible-rabbit-config.yaml` Puoi verificare le nuove configurazioni sulla console RabbitMQ.	RabbitMQ, Amazon MQ, Ansible

Risorse correlate

- Migrazione da RabbitMQ ad Amazon MQ (post sul blog AWS)
- Strumento a riga di comando di gestione (documentazione RabbitMQ)
- Creare o eliminare uno CloudFormation stack AWS (documentazione Ansible)
- Migrazione di applicazioni basate su messaggi su Amazon MQ for RabbitMQ (post sul blog AWS)

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Migliora la qualità delle chiamate sulle postazioni di lavoro degli agenti nei contact center Amazon Connect

Ernest Ozdoba, Amazon Web Services

Riepilogo

I problemi di qualità delle chiamate sono tra i problemi più difficili da risolvere nei contact center. Per evitare problemi di qualità vocale e procedure di risoluzione complesse, è necessario ottimizzare l'ambiente di lavoro e le impostazioni della workstation degli agenti. Questo modello descrive le tecniche di ottimizzazione della qualità vocale per le postazioni di lavoro degli agenti nei contact center Amazon Connect. Fornisce consigli nelle seguenti aree:

- Adeguamenti dell'ambiente di lavoro. L'ambiente in cui si trovano gli agenti non influisce sul modo in cui la voce viene trasmessa sulla rete, ma ha un effetto sulla qualità delle chiamate.
- Impostazioni della workstation dell'agente. Le configurazioni hardware e di rete per le workstation dei contact center hanno effetti significativi sulla qualità delle chiamate.
- Impostazioni del browser. Gli agenti utilizzano un browser Web per accedere al sito Web Amazon Connect Contact Control Panel (CCP) e comunicare con i clienti, in modo che le impostazioni del browser possano influire sulla qualità delle chiamate.

Anche i seguenti componenti possono influire sulla qualità delle chiamate, ma non rientrano nell'ambito della workstation e non rientrano in questo schema:

- Flussi di traffico verso il cloud Amazon Web Services (AWS) tramite AWS Direct Connect, una VPN a tunnel completo o una VPN a tunnel diviso
- Condizioni di rete quando si lavora all'interno o all'esterno della sede aziendale
- Connettività alla rete telefonica pubblica commutata (PSTN)
- Il dispositivo e l'operatore telefonico del cliente
- Configurazione dell'infrastruttura desktop virtuale (VDI)

Per ulteriori informazioni su queste aree, consulta Problemi comuni del Pannello di controllo dei contatti (CCP) e Utilizzo dell'Endpoint Test Utility nella documentazione di Amazon Connect.

Prerequisiti e limitazioni

Prerequisiti

- Le cuffie e le workstation devono soddisfare i requisiti specificati nella Amazon Connect Administrator Guide.

Limitazioni

- Le tecniche di ottimizzazione descritte in questo modello si applicano alla qualità vocale dei soft phone. Non si applicano quando configuri Amazon Connect CCP in modalità telefono fisso. Tuttavia, puoi utilizzare la modalità telefono fisso se la configurazione del softphone non offre una qualità vocale accettabile per la chiamata.

Versioni del prodotto

- Per i browser e le versioni supportati, consulta la Amazon Connect Administrator Guide.

Architecture

Questo modello è indipendente dall'architettura perché si riferisce alle impostazioni delle workstation degli agenti. Come illustrato nel diagramma seguente, il percorso vocale dall'agente al cliente è influenzato dall'auricolare, dal browser, dal sistema operativo, dall'hardware della workstation e dalla rete dell'operatore.

Nei contact center Amazon Connect, la connettività audio dell'utente viene stabilita con WebRTC. La voce è codificata con il codec audio interattivo Opus e crittografata con il Secure Real-time Transport Protocol (SRTP) in transito. Sono possibili altre architetture di rete, tra cui VPN, reti WAN/LAN private e reti ISP.

Tools (Strumenti)

- Amazon Connect Endpoint Test Utility: questa utilità verifica la connettività di rete e le impostazioni del browser.
- Editor di configurazione del browser per le impostazioni WebRTC:
  - Per Firefox: about:config
  - Per Chrome: chrome: //flags
- CCP Log Parser: questo strumento consente di analizzare i log CCP per la risoluzione dei problemi.

Epiche

Modifica l'ambiente di lavoro

Operazione	Description	Competenze richieste
Riduci il rumore di fondo.	Evita ambienti rumorosi. Se ciò non è possibile, ottimizza l'ambiente con questi suggerimenti per l'insonorizzazione: • Assorbe il rumore utilizzando superfici fonoassorbenti come tende, tappeti e mobili imbottiti. • Blocca il rumore inserendo barriere tra le scrivanie. • Prendi in considerazione una soluzione di cancellazione attiva del rumore (ANC) come un generatore di rumore bianco per favorire la concentrazione e garantire la privacy, oppure utilizza cuffie con cancellazione del rumore. • Evita l'eco nelle tue chiamate. Spazi ampi e vuoti potrebbero creare effetti di eco o amplificare i rumori. La copertura di superfici in grado di far rimbalzare i suoni contribuirà a ridurre gli echi.	Agente, manager

Ottimizza le impostazioni della postazione di lavoro degli agenti

Operazione	Description	Competenze richieste
Scegli l'auricolare giusto.	• Se l'ambiente è rumoroso, scegli un auricolare stereo. Indirizzare il suono verso entrambe le orecchie aiuta gli agenti a concentrarsi e a sentire meglio il cliente, e riduce il rumore generale rendendo meno probabile che gli agenti alzino la voce. • Evita di usare altoparlanti o sistemi audio integrati nel computer. Per una qualità ottimale, utilizza una cuffia cablata dedicata all'uso nei contact center. Le cuffie wireless sono comode, ma potrebbero causare un ulteriore ritardo audio e una qualità audio ridotta a causa delle interferenze radio e della transcodifica.	Agente, responsabile
Usa l'auricolare come previsto.	• Attiva le funzioni di cancellazione attiva del rumore e di miglioramento del parlato dell'auricolare, se disponibili. Cerca impostazioni come ANC o ANR. Per istruzioni sull'attivazione di queste impostazioni, consulta il manuale utente dell'auricolare.	Agente

Operazione	Description	Competenze richieste
	• Regola il microfono in modo da poter parlare direttamente al suo interno. La posizione migliore per il microfono è appena sotto il mento. Un posiziona mento corretto può fare una differenza di 10 decibel (dB) nel livello sonoro. La maggior parte delle cuffie consente di ruotare o piegare il braccio del microfono (boom), quindi è importante tenerlo nella posizione giusta quando si parla. • Alcune cuffie sono dotate di più microfoni e di funzional ità avanzate come il voice beamforming, che consente di catturare il parlato senza rumore. Per assicurarti di utilizzare il microfono principale come previsto dal produttore, consulta il manuale utente del tuo dispositivo.	

Operazione	Description	Competenze richieste
Controlla le risorse della workstation.	Assicurati che i computer dei tuoi agenti siano performanti. Se utilizzano applicazioni di terze parti che consumano risorse, i loro computer potrebbero non soddisfare i requisiti hardware minimi per eseguire CCP. Se gli agenti riscontrano problemi di qualità delle chiamate, assicurati evi che abbiano sufficiente potenza di elaborazione (CPU), spazio su disco, larghezza di banda di rete e memoria per CCP. Gli agenti devono chiudere tutte le applicazioni e le schede non necessarie per migliorare le prestazioni del CCP e la qualità delle chiamate.	Amministratore

Operazione	Description	Competenze richieste
Configura le impostazioni audio del sistema operativo.	Le impostazioni predefinite per il livello e l'amplificazione del microfono di solito funzionan o bene. Se riscontri che la voce in uscita è bassa o il microfono capta troppo, potrebbe essere utile modificar e queste impostazioni. Le impostazioni del microfono sono disponibili nella configura zione audio del sistema del computer (Suono, Input su macOS, Proprietà microfono in Windows). Puoi accedere alle impostazioni avanzate che potrebbero influire sulla qualità della voce tramite strumenti di sistema o applicazioni di terze parti. Ecco alcune delle impostazioni che puoi controlla re: • Frequenza di campionam ento: questo valore determina quante volte il suono viene sondato al secondo. L'impostazione predefinita è in genere 44 o 48 kilohertz (kHz). Il valore ottimale per Amazon Connect è 48 kHz. Puoi utilizzare le impostazioni del browser per sostituir e il valore predefinito. Per ulteriori informazioni,	Agente, amministratore

Operazione	Description	Competenze richieste
	consulta la sezione sulla risoluzione dei problemi della Amazon Connect Administrator Guide. • Guadagno: questo valore determina quanto il microfono amplifica il suono. Se aumenti il guadagno, il microfono potrebbe captare più rumore di fondo. • Profondità di bit: questo valore di risoluzione digitale descrive quanti livelli di ampiezza del suono vengono riconosciuti. Maggiore è la profondit à di bit, più fluida sarà la voce. Tuttavia, molte reti di telefonia tradizion ali utilizzano lo standard di modulazione a codice di impulso (PCM), che supporta solo una risoluzio ne a 8 bit. • Soglia aperta: si tratta dell'ampiezza sonora minima percepita da un microfono. Se riscontri problemi di qualità della voce, prova a ripristinare questi valori alle impostazioni predefinite prima di approfond ire.	

Operazione	Description	Competenze richieste
	Per ulteriori informazioni su queste e altre impostazioni regolabili, consulta il manuale del dispositivo.	

Operazione	Description	Competenze richieste
Usa una rete cablata.	In genere, la rete Ethernet cablata ha una latenza inferiore, quindi è più facile fornire la qualità di trasmissi one costante richiesta per la trasmissione di dati vocali. Consigliamo una larghezza di banda minima di 100 KB per chiamata. • Se gli agenti lavorano da casa, consigliamo connessioni cablate o wireless. Non dovrebbero essere necessari più di 150 millisecondi per ascoltare il cliente. Puoi accedere al test di latenza di Amazon Connect dall'Amazon Connect Endpoint Test Utility. Tuttavia, questa utilità misura il ritardo dal browser alle regioni Amazon Connect, non ai clienti. La raccomandazione di un ritardo unidirezionale di 150 millisecondi impedisce all'agente e al cliente di parlare tra loro. Il valore viene misurato da un capo all'altro e ogni elemento aggiunge un ritardo, inclusa la parte della chiamata tra la regione Amazon Connect e il cliente.	Amministratore di rete, agente

Operazione	Description	Competenze richieste
	• Se gli agenti lavorano dall'ufficio, il Wi-Fi aziendale è accettabile purché i parametri rientrino nell'inte rvallo consigliato e il traffico RTP (Real-time Transport Protocol) abbia la priorità.	
Aggiorna i driver hardware.	Quando utilizzi una cuffia USB o un altro tipo di cuffia dotata di un proprio firmware, ti consigliamo di mantenerla aggiornata alla versione più recente. Le cuffie semplici che utilizzano una porta ausiliari a utilizzano il dispositivo audio integrato del computer, quindi assicurati che il driver hardware del sistema operativo sia aggiornato. In rari casi, l'aggiornamento di un driver audio può causare problemi audio e potrebbe essere necessario ripristinarlo. Per ulteriori informazioni sulla modifica delle versioni del firmware e dei driver, consultat e il manuale del dispositivo.	Amministratore

Operazione	Description	Competenze richieste
Evita gli hub e i dongle USB.	Quando colleghi l'auricolare, evita dispositivi aggiuntiv i come dongle, convertitori di tipo porta, hub e cavi di estensione. Questi dispositivi potrebber o influire sulla qualità delle chiamate. Connect invece il dispositivo direttamente alla porta del computer.	Agente

Operazione	Description	Competenze richieste
Controlla i registri CCP.	Il CCP Log Parser offre un modo semplice per controllare i log delle applicazioni. 1. Scaricate i registri CCP dopo una chiamata. 2. Apri CCP Log Parser. 3. Trascina e rilascia il file di registro per caricare il registro per l'analisi. 4. Una volta analizzato il registro, per impostazione predefinita viene seleziona ta la scheda Istantanee e registri. Scegli la scheda Metriche accanto ad essa per controllare le informazi oni dettagliate. 5. Nella sezione WebRTC Metrics - audio_input, controlla quanto segue: • Il grafico del livello audio, per vedere se il livello audio ricevuto è superiore a 0. Ciò indica che l'audio è stato ricevuto dal chiamante. • Il grafico dei pacchetti per tutti i pacchetti persi. Se questo grafico mostra aumenti significativi, contatta il tuo team di supporto IT.	Agente (competenze avanzate)

Operazione	Description	Competenze richieste
	6. Nella sezione WebRTC Metrics - audio_output, controlla quanto segue: • Il grafico del livello audio, per confermare che l'audio è stato inviato dal dispositivo. • Il grafico dei pacchetti. Se notate un picco di perdita di pacchetti, segnalatelo al vostro team di supporto IT. • Il grafico Jitter Buffer & RTT. Valori del tempo di andata e ritorno (RTT) superiori a 300 influiran no sull'esperienza di chiamata. Segnalateli al vostro team di supporto IT.	

Ottimizza le impostazioni del browser

Operazione	Description	Competenze richieste
Ripristina le impostazioni WebRTC predefinite.	WebRTC deve essere abilitato per effettuare chiamate soft phone con CCP. Si consiglia di mantenere le impostazioni predefinite per le funzionalità relative a WebRTC. • In Chrome, puoi impostare i flag accedendo all'URL	Amministratore

Operazione	Description	Competenze richieste
	chrome: //flags. Digita WebRTC nella casella di ricerca per trovare le impostazioni che possono interferire con CCP e impostale su Predefinito. • In Firefox, digita about:con fig nella barra degli indirizzi , quindi digita WebRTC nella casella di ricerca nella pagina di configurazione. Le impostazioni non predefini te vengono visualizzate in grassetto e possono essere modificate in Predefinite.	
Disattiva le estensioni del browser durante la risoluzione dei problemi.	Alcune estensioni del browser potrebbero influire sulla qualità delle chiamate o addirittura impedire il corretto collegame nto delle chiamate. Utilizza la finestra di navigazione in incognito o la modalità privata del browser e disattiva tutte le estensioni. Se questo risolve il problema, esamina le estensioni del browser e cerca componenti aggiuntivi sospetti oppure disattivali singolarm ente.	Agente, amministratore

Operazione	Description	Competenze richieste
Controlla la frequenza di campionamento del browser.	Verifica che l'ingresso del microfono sia impostato sulla frequenza di campionam ento ottimale di 48 kHz. Per istruzioni, consulta la Amazon Connect Administrator Guide.	Agente, amministratore

Risorse correlate

Se hai seguito i passaggi indicati in questo schema ma continui a riscontrare problemi con la qualità delle chiamate, consulta le seguenti risorse per suggerimenti sulla risoluzione dei problemi.

- Esamina i problemi più comuni relativi al Contact Control Panel (CCP).
- Verifica la connessione con l'Endpoint Test Utility.
- Segui la guida alla risoluzione dei problemi per qualsiasi altro problema.

Se la risoluzione dei problemi e le regolazioni non risolvono il problema della qualità delle chiamate, la causa principale potrebbe essere esterna alla workstation. Per ulteriori informazioni sulla risoluzione dei problemi, contattate il team di supporto IT.

Altri modelli

- Scomponi i monoliti in microservizi utilizzando CQRS e l'event sourcing
- Implementa una ChatOps soluzione per gestire i risultati della scansione SAST utilizzando Amazon Q Developer nelle applicazioni di chat, azioni personalizzate e CloudFormation
- Integra Amazon API Gateway con Amazon SQS per gestire REST asincrono APIs
- Semplifica lo sviluppo e la distribuzione di bot Amazon Lex utilizzando un flusso di lavoro automatizzato

Strategia multi-account

Argomenti

- Esegui la migrazione di un account AWS membro da a AWS Organizations AWS Control Tower
- Imposta avvisi per la chiusura programmatica degli account in AWS Organizations
- Altri modelli

Esegui la migrazione di un account AWS membro da a AWS Organizations AWS Control Tower

Rodolfo Junior Cerrada, Amazon Web Services

Riepilogo

Questo modello descrive come migrare Account AWS da AWS Organizations un account membro a un account di gestione verso. AWS Control Tower Registrando l'account AWS Control Tower, puoi sfruttare i controlli e le funzionalità preventivi e investigativi che semplificano la governance dell'account. Potresti anche voler migrare il tuo account membro se il tuo account di gestione AWS Organizations è stato compromesso e desideri trasferire gli account membro in una nuova organizzazione governata da. AWS Control TowerAWS Control Tower

AWS Control Tower fornisce un framework che combina e integra le funzionalità di molti altri Servizi AWS AWS Organizations, tra cui e garantisce conformità e governance coerenti in tutto l'ambiente multi-account. Con AWS Control Tower, puoi seguire una serie di regole e definizioni prescritte che estendono le funzionalità di. AWS Organizations Ad esempio, è possibile utilizzare i controlli per garantire che i registri di sicurezza e le necessarie autorizzazioni di accesso tra account vengano creati e non modificati.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- AWS Control Tower configurato nell'organizzazione di destinazione in AWS Organizations (per istruzioni, consulta Configurazione nella AWS Control Tower documentazione)
- Credenziali di amministratore per AWS Control Tower (membro del AWSControlTowerAdminsgruppo)
- Credenziali di amministratore per la fonte Account AWS

Limitazioni

- L'account di gestione di origine in AWS Organizations deve essere diverso dall'account di gestione di destinazione in AWS Control Tower.

Versioni del prodotto

- AWS Control Tower versione 2.3 (febbraio 2020) o successiva (vedi note di rilascio)

Architecture

Il diagramma seguente illustra il processo di migrazione e l'architettura di riferimento. Questo modello migra il Account AWS dall'organizzazione di origine a un'organizzazione di destinazione governata da. AWS Control Tower

Il processo di registrazione consiste nei seguenti passaggi:

1. L'organizzazione di destinazione invia un invito affinché l'account entri a far parte dell'organizzazione.
2. L'account accetta l'invito e diventa membro dell'organizzazione target.
3. L'account viene registrato AWS Control Tower e trasferito in un'unità organizzativa (OU) registrata. (Ti consigliamo di controllare la AWS Control Tower dashboard per confermare l'iscrizione.) A questo punto, tutti i controlli abilitati nell'unità organizzativa registrata hanno effetto.

Tools (Strumenti)

Servizi AWS

- AWS Organizationsè un servizio di gestione degli account che consente di consolidare più account Account AWS in un'unica entità (un'organizzazione) da creare e gestire centralmente.
- AWS Control Towerintegra le funzionalità di altri servizi, tra cui, e AWS Organizations AWS IAM Identity Center AWS Service Catalog, per aiutarvi a far rispettare e gestire le regole di governance per la sicurezza, le operazioni e la conformità su larga scala in tutte le organizzazioni e gli account del. Cloud AWS

Epiche

Invita l'account a entrare a far parte della nuova organizzazione con AWS Control Tower

Operazione	Description	Competenze richieste
Accedi a AWS Control Tower.	Accedi alla AWS Control Tower console come amministratore. Attualmente, non esiste un modo diretto per spostare un uomo Account AWS da un'organizzazione di origine a un'organizzazione in un'unità organizzativa governata da AWS Control Tower. Tuttavia, è possibile estendere la AWS Control Tower governance a un'organizzazione esistente Account AWS quando la si registra in un'unità organizza tiva già governata da AWS Control Tower. Ecco perché è necessario accedere a AWS Control Tower questa fase.	Amministratore di AWS Control Tower
Invita l'account membro.	1. Accedi alla AWS Organizat ions console e vai alla Account AWSpagina. 2. Nella pagina Aggiungi una Account AWS pagina, scegli Invita un esistente Account AWS. 3. Completa le informazi oni sull'account, incluso il numero di account a 12	Amministratore di AWS Control Tower

Operazione	Description	Competenze richieste
	cifre (senza trattini) e la descrizione e i tag opzionali , quindi scegli Invia invito. **Important** Verifica che il trasferimento dell'account non influisca sulle applicazioni o sulla connettività di rete. Questa azione invia un'e-mail di invito con un link all'account del membro. Quando l'amministratore dell'account segue il link e accetta l'invito, l'account del membro viene visualizzato nella Account AWSpagina. Per ulteriori informazioni, consulta Gestire gli inviti all'account nella AWS Organizations documentazione.	

Operazione	Description	Competenze richieste
Testa applicazioni e connettività.	Quando l'account del membro è stato registrato nella nuova organizzazione, viene visualizzato nell'unità organizzativa all'interno di una cartella root. Viene inoltre visualizzato nella AWS Control Tower console, contrassegnato come non registrato negli account, perché non è ancora stato registrato nell'unità organizzativa registrata. AWS Control Tower Verificare quanto segue: • Controlla la AWS Control Tower dashboard per vedere se ci sono violazioni del guardrail. • Controlla la connettività di rete (VPN o AWS Direct Connect) per assicurarti che non sia stata influenzata dal trasferimento. • (Proprietari delle applicazioni) Testa le applicazioni associate a questo account per verificare che funzionino come previsto e che le dipendenze non siano state influenzate dal trasferimento dell'account.	Amministratore AWS Control Tower, amministratore dell'account membro, proprietari delle applicazioni

Prepara l'account per la registrazione

Operazione	Description	Competenze richieste
Rivedi i controlli e correggi eventuali violazioni.	Rivedi i controlli definiti nell'unità organizzativa di destinazione, in particolare i controlli preventivi, e correggi eventuali violazioni. Una serie di controlli preventivi obbligatori sono abilitati di default quando configuri la AWS Control Tower landing zone. Questi non possono essere disabilitati. È necessario rivedere questi controlli obbligatori e correggere l'account del membro (manualmente o utilizzando uno script) prima di registrare l'account. **Note** I controlli preventivi mantengono gli account AWS Control Tower registrati conformi e prevengono le violazioni delle politiche. Qualsiasi violazione dei controlli preventivi potrebbe influire sull'iscrizione. Le violazioni del controllo investigativo	Amministratore di AWS Control Tower, amministratore dell'account dei membri

Operazione	Description	Competenze richieste
	vengono visualizzate nella AWS Control Tower dashboard , se rilevate, dopo l'avvenuta registraz ione. Non influisco no sul processo di registrazione. Per ulteriori informazioni, consulta Informazi oni sui controlli nella AWS Control Tower documentazione.	
Verifica la presenza di problemi di connettività dopo aver corretto le violazioni di controllo.	In alcuni casi, potrebbe essere necessario chiudere porte specifiche o disabilitare servizi per correggere le violazion i di controllo. Assicurati che le applicazioni che utilizzan o tali porte e servizi vengano corrette prima di registrare l'account.	Proprietario dell'applicazione

Registra l'account in AWS Control Tower

Operazione	Description	Competenze richieste
Accedi a AWS Control Tower.	Accedi alla console AWS Control Tower. Utilizza credenziali di accesso con autorizzazioni amministrative per. AWS Control Tower Non utilizzare le credenzia	Amministratore di AWS Control Tower

Operazione	Description	Competenze richieste
	li dell'utente root (account di gestione) per registrare un account. AWS Organizations Verrà visualizzato un messaggio di errore.	
Registra l'account.	1. Dalla pagina Account Factory in AWS Control Tower, scegli Registra account. 2. Inserisci i dettagli, tra cui l'indirizzo e-mail associato all'account che desideri registrare, il nome visualizzato in cui verrà visualizzato AWS Control Tower, l'indirizzo e-mail di IAM Identity Center, il nome e cognome del proprietario dell'account e l'unità organizzativa in cui desideri registrare l'account. L'indirizzo e-mail di IAM Identity Center è l'indirizzo e-mail utente preferito. Puoi utilizzare lo stesso indirizzo e-mail dell'account. 3. Scegli Enroll account (Registra account). Per ulteriori informazioni, consulta Informazioni sulla registrazione degli account esistenti nella AWS Control Tower documentazione.	Amministratore di AWS Control Tower

Verifica l'account dopo l'iscrizione

Operazione	Description	Competenze richieste
Verifica l'account.	Da AWS Control Tower, scegli Account. L'account che hai appena registrato ha lo stato iniziale di Iscrizione. Una volta completata l'iscrizione, lo stato cambia in Registrato.	Amministratore di AWS Control Tower, amministratore dell'account dei membri
Verifica la presenza di violazioni di controllo.	I controlli definiti nell'unit à organizzativa verranno applicati automaticamente all'account membro registrat o. Monitora la AWS Control Tower dashboard per rilevare eventuali violazioni e correggil e di conseguenza. Per ulteriori informazioni, consulta Informazioni sui controlli nella AWS Control Tower documentazione.	Amministratore di AWS Control Tower, amministratore dell'account dei membri

risoluzione dei problemi

Problema	Soluzione
Riceverai il messaggio di errore: Si è verificat o un errore sconosciuto. Riprova più tardi o contatta l' AWS assistenza.	Questo errore si verifica quando si utilizzan o le credenziali dell'utente root (account di gestione) AWS Control Tower per registrare un nuovo account. AWS Service Catalog non riesce a mappare il portafoglio o il prodotto Account Factory all'utente root, il che genera il messaggio di errore. Per correggere questo errore, utilizza credenziali utente (amministrat

Problema	Soluzione
	ratore) non root con accesso completo per registrare il nuovo account. Per ulteriori informazioni su come assegnare l'accesso amministrativo a un utente amministrativo, consulta Guida introduttiva nella documenta zione di IAM Identity Center.
La pagina AWS Control Tower Attività mostra un'azione Get Catastrophic Drift.	Questa azione riflette un controllo della situazione del servizio e non indica alcun problema con la configurazione. AWS Control Tower Nessuna operazione richiesta.

Risorse correlate

Documentazione

- Terminologia e concetti (documentazione)AWS Organizations
- Che cos'è AWS Control Tower? (AWS Control Tower documentazione)
- Rimuovere un account membro da un'organizzazione (AWS Organizations documentazione)
- Configurazione (AWS Control Tower documentazione)

Tutorial e video

- AWS Control Tower workshop (workshop autogestito)
- Che cos'è? AWS Control Tower (video)
- Assegnazione di ruoli agli utenti in AWS Control Tower (video)

Imposta avvisi per la chiusura programmatica degli account in AWS Organizations

Richard Milner-Watts, Debojit Bhadra e Manav Yadav, Amazon Web Services

Riepilogo

L'CloseAccount API for AWS Organizations consente di chiudere gli account dei membri all'interno di un'organizzazione in modo programmatico, senza dover accedere all'account con credenziali root. L'RemoveAccountFromOrganization API estrae un account da un'organizzazione in AWS Organizations, quindi diventa un account autonomo.

Questi APIs potrebbero aumentare il numero di operatori che possono chiudere o rimuovere un account AWS. Tutti gli utenti che hanno accesso all'organizzazione tramite AWS Identity and Access Management (IAM) nell'account di gestione AWS Organizations possono richiamarli APIs, quindi l'accesso non è limitato al proprietario dell'e-mail root dell'account con qualsiasi dispositivo di autenticazione a più fattori (MFA) associato.

Questo modello implementa avvisi quando `RemoveAccountFromOrganization` APIs vengono chiamati i `CloseAccount` e, quindi, è possibile monitorare queste attività. Per gli avvisi, utilizza un argomento Amazon Simple Notification Service (Amazon SNS). Puoi anche configurare le notifiche Slack tramite un webhook.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un'organizzazione in AWS Organizations
- Accesso all'account di gestione dell'organizzazione, nella directory principale dell'organizzazione, per creare le risorse necessarie

Limitazioni

- Come descritto nel riferimento all'API di AWS Organizations, l'`CloseAccount`API consente di chiudere solo il 10% degli account dei membri attivi entro un periodo continuativo di 30 giorni.

- Quando un account AWS viene chiuso, il suo stato viene modificato in SOSPESO. Per 90 giorni dopo questa transizione di status, AWS Support può riaprire l'account. Dopo 90 giorni l'account viene eliminato definitivamente.
- Utenti che hanno accesso all'account di gestione di AWS Organizations e APIs potrebbero anche disporre delle autorizzazioni per disabilitare questi avvisi. Se la preoccupazione principale è il comportamento dannoso anziché l'eliminazione accidentale, prendi in considerazione la possibilità di proteggere le risorse create da questo modello con un limite di autorizzazioni IAM.
- L'API richiede `CloseAccount` e `RemoveAccountFromOrganization` viene elaborata nella regione Stati Uniti orientali (Virginia settentrionale) (). `us-east-1` Pertanto, è necessario implementare questa soluzione per osservare gli eventi. `us-east-1`

Architecture

Stack tecnologico Target

- AWS Organizations
- AWS CloudTrail
- Amazon EventBridge
- AWS Lambda
- Amazon SNS

Architettura Target

Il diagramma seguente mostra l'architettura della soluzione per questo modello.

1. AWS Organizations elabora una `RemoveAccountFromOrganization` richiesta `CloseAccount` or.
2. Amazon EventBridge è integrato con AWS CloudTrail per inviare questi eventi al bus eventi predefinito.
3. Una EventBridge regola Amazon personalizzata corrisponde alle richieste di AWS Organizations e chiama una funzione AWS Lambda.
4. La funzione Lambda invia un messaggio a un argomento SNS, a cui gli utenti possono iscriversi per ricevere avvisi e-mail o ulteriori elaborazioni.

5. Se le notifiche Slack sono abilitate, la funzione Lambda invia un messaggio a un webhook Slack.

Tools (Strumenti)

Servizi AWS

- AWS CloudFormation offre un modo per modellare una raccolta di risorse AWS e di terze parti correlate, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita, trattando l'infrastruttura come codice.
- Amazon EventBridge è un servizio di bus eventi senza server che puoi utilizzare per connettere le tue applicazioni con dati provenienti da una varietà di fonti. EventBridge riceve un evento, un indicatore di un cambiamento nell'ambiente e applica una regola per indirizzare l'evento verso un obiettivo. Le regole abbinano gli eventi agli obiettivi in base alla struttura dell'evento, chiamata pattern di evento, o a una pianificazione.
- AWS Lambda è un servizio di elaborazione che supporta l'esecuzione di codice senza effettuare il provisioning o la gestione di server. Lambda esegue il codice solo quando necessario e si ridimensiona automaticamente, da poche richieste al giorno a migliaia al secondo. I costi saranno calcolati in base al tempo di elaborazione effettivo. Quando il codice non è in esecuzione non viene addebitato alcun costo.
- AWS Organizations ti aiuta a gestire e governare centralmente il tuo ambiente man mano che cresci e ridimensioni le tue risorse AWS. Con AWS Organizations, puoi creare in modo programmatico nuovi account AWS e allocare risorse, raggruppare account per organizzare i flussi di lavoro, applicare policy ad account o gruppi per la governance e semplificare la fatturazione utilizzando un unico metodo di pagamento per tutti i tuoi account.
- AWS CloudTrail monitora e registra l'attività degli account nell'infrastruttura AWS e ti offre il controllo sulle azioni di storage, analisi e correzione.
- Amazon Simple Notification Service (Amazon SNS) è un servizio di messaggistica completamente gestito per application-to-application le comunicazioni (A2A) e (A2P) application-to-person.

Altri strumenti

- La libreria AWS Lambda Powertools for Python è un insieme di utilità che forniscono funzionalità di tracciamento, registrazione, metriche e gestione degli eventi per le funzioni Lambda.

Codice

Il codice per questo pattern si trova nel repository GitHub AWS Account Closer Notifier.

La soluzione include un CloudFormation modello che implementa l'architettura per questo pattern. Utilizza la libreria AWS Lambda Powertools for Python per fornire registrazione e tracciamento.

Epiche

Implementa l'architettura

Operazione	Description	Competenze richieste
Avvia il CloudFormation modello per lo stack di soluzioni.	Il CloudFormation modello per questo modello si trova nel ramo principale del GitHub repository. Implementa i ruoli, EventBridge le regole, le funzioni Lambda e l'argomento SNS di IAM. Per avviare il modello: 1. Clona il GitHub repository per ottenere una copia del codice della soluzione. 2. Apri la Console di gestione AWS per l'account di gestione AWS Organizations. 3. Scegli la regione Stati Uniti orientali (Virginia settentrionale) (`us-east-1`), quindi apri la CloudFormation console. 4. Create lo stack utilizzando il `account-closure-notifier.yml` modello e specificando i seguenti valori:	Amministratore AWS

Operazione	Description	Competenze richieste
	• Nome stack: `aws-account-closure-notifier-stack` • `ResourceP refix` parametro: `aws-account-closure-notifier` • `SlackNoti fication` parametro: Se sono necessarie notifiche Slack, modifica questa impostazione in `true`. • `SlackWebhookEndpoi nt` parametro: Se sono necessarie le notifiche Slack, specifica l'URL del webhook. Per ulteriori informazioni sul lancio di uno CloudFormation stack, consulta la documenta zione AWS.	

Operazione	Description	Competenze richieste
Verifica che la soluzione sia stata avviata correttamente.	1. Attendi che lo CloudFormation stack raggiunga lo stato CREATE_COMPLETE. 2. Apri la console in. EventBridge `us-east-1` 3. Verifica che sia stata creata una nuova regola con il nome`aws-account-closure-notifier-event-rule` .	Amministratore AWS

Operazione	Description	Competenze richieste
Iscriviti all'argomento SNS.	(Facoltativo) Se desideri iscriverti all'argomento SNS: 1. Apri la console Amazon SNS in `us-east-1` e trova l'argomento denominato. `aws-account-closure-notifier-sns-topic` 2. Scegli il nome dell'argomento, quindi scegli Crea abbonamento. 3. Per Protocollo, scegli E-mail. 4. Per Endpoint, specifica l'indirizzo email che deve ricevere la notifica, quindi scegli Crea abbonamento. 5. Controlla la tua casella di posta elettronica per ricevere un messaggio da AWS Notifications. Utilizza il link contenuto in questa e-mail per confermare l'iscrizione. Per ulteriori informazioni sulla configurazione delle notifiche SNS, consulta la documentazione di Amazon SNS.	Amministratore AWS

Verifica la soluzione

Operazione	Description	Competenze richieste
Invia un evento di test al bus eventi predefinito.	Il GitHub repository fornisce un esempio di evento che è possibile inviare al bus di eventi EventBridge predefinito per il test. La EventBridge regola reagisce anche agli eventi che utilizzano l'origine degli eventi personalizzata. `account.closure.notifier` **Note** Non puoi utilizzare l'origine CloudTrail dell'evento per inviare questo evento, perché non è possibile inviare un evento come servizio AWS. Per inviare un evento di test: 1. Apri la EventBridge console in `us-east-1` . 2. Nel pannello di navigazione, in Bus, scegli Event bus, quindi seleziona il bus eventi predefinito. 3. Scegli Invia eventi.	Amministratore AWS

Operazione	Description	Competenze richieste
	4. Per Origine dell'evento, inserisci `account.closure.notifier` . 5. In Tipo di dettaglio, immetti `AWS API Call via CloudTrail` . 6. Per informazioni dettagliate sull'evento, copia e incolla il contenuto di `tests/dummy-event.json` dal GitHub repository nella casella di testo. 7. Scegliete Invia per avviare il flusso di lavoro delle notifiche.	
Verifica che la notifica via e-mail sia stata ricevuta.	Controlla la casella di posta che ha sottoscritto l'argomento SNS per le notifiche. Dovresti ricevere un'e-mail con i dettagli dell'account che è stato chiuso e del principale che ha eseguito la chiamata API.	Amministratore AWS

Operazione	Description	Competenze richieste
Verifica che la notifica Slack sia stata ricevuta.	(Facoltativo) Se hai specifica to un URL del webhook per il `SlackWebhookEndpoi nt` parametro quando hai distribuito il CloudFormation modello, controlla il canale Slack mappato al webhook. Dovrebbe visualizzare un messaggio con i dettagli dell'account che è stato chiuso e del principale che ha eseguito la chiamata API.	Amministratore AWS

Risorse correlate

- CloseAccount azione (riferimento all'API AWS Organizations)
- RemoveAccountFromOrganization azione (riferimento all'API AWS Organizations)
- AWS Lambda Powertools per Python

Altri modelli

- Automatizza la creazione di account utilizzando Landing Zone Accelerator su AWS
- Eliminazione automatica degli AWS CloudFormation stack e delle risorse associate
- Automatizza la gestione dinamica delle pipeline per l'implementazione di soluzioni hotfix in ambienti Gitflow utilizzando e AWS Service Catalog AWS CodePipeline
- Crea una rete dati aziendale con Amazon DataZone e AWS CDK AWS CloudFormation
- Centralizza il monitoraggio utilizzando Amazon CloudWatch Observability Access Manager
- Gestisci i set di autorizzazioni per più account utilizzando Account Factory for Terraform
- Implementa una strategia di ramificazione Gitflow per ambienti con più account DevOps
- Implementa una strategia di ramificazione GitHub Flow per ambienti con più account DevOps
- Implementa una strategia di ramificazione Trunk per ambienti con più account DevOps
- Gestisci i set di AWS autorizzazioni in modo dinamico utilizzando Terraform
- Crea un'architettura IPAM gerarchica e multiregionale utilizzando Terraform AWS
- Imposta il rilevamento della CloudFormation deriva in un'organizzazione multiregionale e con più account
- Configura la risoluzione DNS per reti ibride in un ambiente multi-account AWS

Sicurezza, identità e conformità

Argomenti

- Automatizza la risposta agli incidenti e l'analisi forense
- Controlla automaticamente i gruppi AWS di sicurezza che consentono l'accesso da indirizzi IP pubblici
- Automatically remediate unencrypted Amazon RDS DB instances and clusters
- Convalida e distribuisci automaticamente le policy e i ruoli IAM utilizzando CodePipeline IAM Access Analyzer e le macro AWS CloudFormation
- Integrazione bidirezionale AWS Security Hub CSPM con il software Jira
- Crea una pipeline per immagini di container rinforzate utilizzando Image EC2 Builder e Terraform
- Centralizza la gestione delle chiavi di accesso IAM in AWS Organizations utilizzando Terraform
- Controlla una CloudFront distribuzione Amazon per la registrazione degli accessi, la versione HTTPS e TLS
- Scegli un flusso di autenticazione Amazon Cognito per applicazioni aziendali
- Crea regole AWS Config personalizzate utilizzando AWS CloudFormation Guard le politiche
- Crea un report consolidato dei risultati di sicurezza di Prowler da più Account AWS
- Implementa e gestisci AWS Control Tower i controlli utilizzando AWS CDK e CloudFormation
- Implementa e gestisci AWS Control Tower i controlli utilizzando Terraform
- Implementa le automazioni di sicurezza per la AWS WAF soluzione utilizzando Terraform
- Implementa una pipeline che rilevi simultaneamente i problemi di sicurezza in più risultati di codice
- Implementa controlli di accesso basati su attributi investigativi per sottoreti pubbliche utilizzando AWS Config
- Implementa controlli di accesso preventivi basati sugli attributi per le sottoreti pubbliche
- Rileva le istanze di database Amazon RDS e Aurora con certificati CA in scadenza
- Genera dinamicamente una policy IAM con IAM Access Analyzer utilizzando Step Functions
- Abilita Amazon in GuardDuty modo condizionale utilizzando modelli AWS CloudFormation
- Abilita la crittografia trasparente dei dati in Amazon RDS for SQL Server
- Monitora e correggi l'eliminazione pianificata delle chiavi AWS KMS
- Identifica i bucket Amazon S3 pubblici utilizzando Security Hub (AWS Organizations CSPM)
- Inserisci e analizza i log di sicurezza AWS in Microsoft Sentinel

- Gestisci AWS Organizations le policy come codice utilizzando AWS CodePipeline Amazon Bedrock
- Gestisci i set di AWS IAM Identity Center autorizzazioni come codice utilizzando AWS CodePipeline
- Gestisci le credenziali con AWS Secrets Manager
- Monitora ElastiCache i cluster Amazon per la crittografia a riposo
- Monitoraggio dell'attività dell'utente root IAM
- Invia una notifica quando viene creato un utente IAM
- Impedisci l'accesso a Internet a livello di account utilizzando una politica di controllo del servizio
- Esporta un report delle identità di AWS IAM Identity Center e delle relative assegnazioni utilizzando PowerShell
- Limita l'accesso in base all'indirizzo IP o alla geolocalizzazione utilizzando AWS WAF
- Scansiona gli archivi Git alla ricerca di informazioni sensibili e problemi di sicurezza utilizzando git-secrets
- Trasferimenti sicuri di file utilizzando Transfer Family, Amazon Cognito e GuardDuty
- Proteggi i dati sensibili nei CloudWatch registri utilizzando Amazon Macie
- Invia avvisi da AWS Network Firewall a un canale Slack
- Invia attributi personalizzati ad Amazon Cognito e inseriscili in token
- Semplifica la gestione privata dei certificati utilizzando AWS Private CA e AWS RAM
- Semplifica la gestione EC2 della conformità di Amazon con agenti Amazon Bedrock e AWS Config
- Aggiorna le credenziali dell'interfaccia a riga di comando AWS da AWS IAM Identity Center utilizzando PowerShell
- Usa Network Firewall per acquisire i nomi di dominio DNS dall'Indicazione del nome del server per il traffico in uscita
- Usa Terraform per abilitare automaticamente Amazon GuardDuty per un'organizzazione
- Verifica le best practice operative per PCI DSS 4.0 utilizzando AWS Config
- Altri modelli

Automatizza la risposta agli incidenti e l'analisi forense

Lucas Kauffman e Tomek Jakubowski, Amazon Web Services

Riepilogo

Questo modello implementa una serie di processi che utilizzano funzioni per fornire quanto segue: AWS Lambda

- Un modo per avviare il processo di risposta agli incidenti con una conoscenza minima
- Processi automatizzati e ripetibili, allineati alla Security Incident Response Guide AWS
- Separazione degli account per gestire le fasi di automazione, archiviare gli artefatti e creare ambienti forensi

Il framework Automated Incident Response and Forensics segue un processo forense digitale standard composto dalle seguenti fasi:

1. Contenimento
2. Acquisizione
3. Esame
4. Analisi

È possibile eseguire indagini su dati statici (ad esempio, memoria acquisita o immagini del disco) e su dati dinamici attivi ma su sistemi separati.

Per ulteriori dettagli, consulta la sezione Informazioni aggiuntive.

Prerequisiti e limitazioni

Prerequisiti

- Due Account AWS:
  - Account di sicurezza, che può essere un account esistente, ma è preferibilmente nuovo
  - Account forense, preferibilmente nuovo
- AWS Organizations configurare
- Negli account dei membri di Organizations:

  - Il ruolo Amazon Elastic Compute Cloud (Amazon EC2) deve avere accesso Get and List ad Amazon Simple Storage Service (Amazon S3) ed essere accessibile da. AWS Systems Manager Consigliamo di utilizzare il ruolo `AmazonSSMManagedInstanceCore` AWS gestito. Tieni presente che questo ruolo verrà automaticamente associato all' EC2 istanza Amazon quando viene avviata la risposta all'incidente. Al termine della risposta, AWS Identity and Access Management (IAM) rimuoverà tutti i diritti sull'istanza.
  - Endpoint del cloud privato virtuale (VPC) nell'account AWS membro e nella risposta e analisi degli incidenti. VPCs Questi endpoint sono: S3 Gateway, EC2 Messages, SSM e SSM Messages.
- AWS Command Line Interface (AWS CLI) installato sulle EC2 istanze Amazon. Se le EC2 istanze Amazon non sono AWS CLI installate, sarà necessario l'accesso a Internet per il corretto funzionamento dello snapshot del disco e dell'acquisizione della memoria. In questo caso, gli script si collegheranno a Internet per scaricare i file di AWS CLI installazione e li installeranno sulle istanze.

Limitazioni

- Questo framework non intende generare artefatti che possano essere considerati prove elettroniche, ammissibili in tribunale.
- Attualmente, questo modello supporta solo istanze basate su Linux in esecuzione su architettura x86.

Architecture

Architettura Target

Oltre all'account membro, l'ambiente di destinazione è composto da due account principali: un account Security e un account Forensics. Vengono utilizzati due account per i seguenti motivi:

- Per separarli dagli account di qualsiasi altro cliente per ridurre il raggio di esplosione in caso di esito negativo dell'analisi forense
- Per contribuire a garantire l'isolamento e la protezione dell'integrità degli artefatti analizzati
- Per mantenere riservate le indagini
- Per evitare situazioni in cui gli autori delle minacce potrebbero aver utilizzato tutte le risorse immediatamente disponibili, si è compromessi Account AWS superando le quote di servizio e impedendoti così di creare un'istanza EC2 Amazon per eseguire indagini.

Inoltre, disporre di account di sicurezza e forensi separati consente di creare ruoli separati: un risponditore per l'acquisizione delle prove e un investigatore per l'analisi. Ogni ruolo avrebbe accesso al proprio account separato.

Il diagramma seguente mostra solo l'interazione tra gli account. I dettagli di ciascun account sono mostrati nei diagrammi successivi e viene allegato un diagramma completo.

Il diagramma seguente mostra l'account del membro.

1. Viene inviato un evento all'argomento Slack Amazon Simple Notification Service (Amazon SNS).

Il diagramma seguente mostra l'account Security.

2. L'argomento Amazon SNS nell'account Security avvia gli eventi Forensics.

Il diagramma seguente mostra l'account Forensics.

L'account Security è il luogo in cui vengono creati i due AWS Step Functions flussi di lavoro principali per l'acquisizione della memoria e dell'immagine del disco. Dopo l'esecuzione dei flussi di lavoro, accedono all'account membro in cui sono coinvolte EC2 le istanze Amazon in un incidente e avviano una serie di funzioni Lambda che raccoglieranno un dump della memoria o un dump del disco. Questi artefatti vengono quindi archiviati nell'account Forensics.

L'account Forensics conterrà gli artefatti raccolti dal flusso di lavoro Step Functions nel bucket Analysis Artifacts Amazon S3. L'account Forensics avrà anche una pipeline Amazon Image EC2 Builder che crea un'Amazon Machine Image (AMI) di un'istanza Forensics. Attualmente, l'immagine è basata su SANS SIFT Workstation.

Il processo di compilazione utilizza il VPC di manutenzione, che dispone di connettività a Internet. L'immagine può essere successivamente utilizzata per avviare l' EC2 istanza Amazon per l'analisi degli artefatti raccolti nell'Analysis VPC.

Analysis VPC non dispone di connettività Internet. Per impostazione predefinita, il pattern crea tre sottoreti di analisi private. È possibile creare fino a 200 sottoreti, ovvero la quota per il numero di sottoreti in un VPC, ma è necessario aggiungere tali sottoreti agli endpoint VPC per automatizzare l'esecuzione dei comandi al loro interno. AWS Systems Manager Session Manager

Dal punto di vista delle best practice, consigliamo di utilizzare ed eseguire le seguenti operazioni: AWS CloudTrail AWS Config

- Tieni traccia delle modifiche apportate nel tuo account Forensics
- Monitora l'accesso e l'integrità degli artefatti archiviati e analizzati

Flusso di lavoro

Il diagramma seguente mostra i passaggi chiave di un flusso di lavoro che include il processo e l'albero decisionale, dal momento in cui un'istanza viene compromessa fino all'analisi e al contenimento.

1. Il `SecurityIncidentStatus` tag è stato impostato con il valore Analyze? In caso affermativo, procedi come segue:
   a. Allega i profili IAM corretti per AWS Systems Manager Amazon S3.
   b. Invia un messaggio Amazon SNS alla coda Amazon SNS in Slack.
   c. Invia un messaggio Amazon SNS alla `SecurityIncident` coda.
   d. Richiama la macchina a stati di acquisizione della memoria e del disco.
2. Memoria e disco sono stati acquisiti? Se no, c'è un errore.
3. Etichetta l' EC2 istanza Amazon con il `Contain` tag.
4. Collega il ruolo IAM e il gruppo di sicurezza per isolare completamente l'istanza.

Automazione e scalabilità

L'intento di questo modello è fornire una soluzione scalabile per eseguire la risposta agli incidenti e l'analisi forense su più account all'interno di una singola organizzazione. AWS Organizations

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationti aiuta a configurare AWS le risorse, a fornirle in modo rapido e coerente e a gestirle durante il loro ciclo di vita in tutte le regioni. Account AWS
- AWS Command Line Interface (AWS CLI) è uno strumento open source con cui interagire Servizi AWS tramite comandi nella shell della riga di comando.

- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Security Hub CSPMfornisce una visione completa dello stato di sicurezza in. AWS Inoltre, consente di verificare la conformità AWS dell'ambiente agli standard e alle best practice del settore della sicurezza.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.
- AWS Step Functionsè un servizio di orchestrazione senza server che ti aiuta a combinare AWS Lambda funzioni e altro per creare applicazioni aziendali critiche. Servizi AWS
- AWS Systems Managerconsente di gestire le applicazioni e l'infrastruttura in esecuzione in. Cloud AWS Semplifica la gestione delle applicazioni e delle risorse, riduce i tempi di rilevamento e risoluzione dei problemi operativi e aiuta a gestire le AWS risorse in modo sicuro su larga scala.

Codice

Per il codice e le indicazioni specifiche sull'implementazione e l'utilizzo, consulta il repository GitHub Automated Incident Response and Forensics Framework.

Epiche

Distribuisci i modelli CloudFormation

Operazione	Description	Competenze richieste
Implementa CloudFormation modelli.	I CloudFormation modelli sono contrassegnati da 1 a 7 con la prima parola del nome dello script che indica in quale account il modello	Amministratore AWS

Operazione	Description	Competenze richieste
	deve essere distribuito. Tieni presente che l'ordine di avvio dei CloudFormation modelli è importante. • `1-forensic-AnalysisVPCnS3Buckets.yaml` : Implementato nell'account forense. Crea i bucket Amazon S3 e l'Analysis VPC e si attiva. CloudTrail • `2-forensic-MaintenanceVPCnEC2ImageBuilderPipeline.yaml` : implementa la pipeline di manutenzione VPC e image builder basata su SANS SIFT. • `3-security_IR-Disk_Mem_automation.yaml` : implementa le funzioni nell'account di sicurezza che consentono l'acquisizione di dischi e memoria. • `4-security_LiME_Volatility_Factory.yaml` : avvia una funzione di compilazione per iniziare a creare i moduli di memoria basati sull'AMI IDs specificata. Nota che IDs gli AMI sono diversi Regioni AWS. Ogni volta che hai bisogno di nuovi moduli di memoria,	

Operazione	Description	Competenze richieste
	puoi eseguire nuovamente questo script con la nuova AMI. IDs Valuta la possibili tà di integrarlo con le tue pipeline Golden Image AMI Builder (se utilizzate nel tuo ambiente). • `5-member-IR-automation.yaml` : Crea la funzione di automazione della risposta agli incidenti del membro, che avvia il processo di risposta agli incidenti. Consente la condivisione dei volumi di Amazon Elastic Block Store (Amazon EBS) tra account, la pubblicazione automatica sui canali Slack durante il processo di risposta agli incidenti, l'avvio del processo di analisi forense e l'isolamento delle istanze al termine del processo. • `6-forensic-artifact-s3-policies.yaml` : Dopo che tutti gli script sono stati distribuiti, questo script corregge le autorizzazioni richieste per tutte le interazioni tra account. • `7-security-IR-vpc.yaml` : configura un VPC utilizzato per l'elaborazione	

Operazione	Description	Competenze richieste
	del volume di risposta agli incidenti. Per avviare il framework di risposta agli incidenti per un' EC2 istanza Amazon specifica , crea un tag con la chiave `SecurityIncidentStatus` e il valore`Analyze`. Ciò avvierà la funzione membro Lambda che avvierà automaticamente l'isolamento, la memoria e l'acquisizione del disco.	
Gestisci il framework.	La funzione Lambda rietichetterà la risorsa anche alla fine (o in caso di errore) con. `Contain` Ciò avvia il contenimento, che isola completamente l'istanza senza un gruppo di INBOUND/OUTBOUND sicurezza e con un ruolo IAM che non consente tutti gli accessi. Segui i passaggi indicati nel repository. GitHub	Amministratore AWS

Implementa azioni CSPM personalizzate di Security Hub

Operazione	Description	Competenze richieste
Distribuisci le azioni CSPM personalizzate di Security	Per creare un'azione personalizzata in modo da poter	Amministratore AWS

Operazione	Description	Competenze richieste
Hub utilizzando un modello. CloudFormation	utilizzare l'elenco a discesa di Security Hub CSPM, distribui sci il modello. `Modules/SecurityHub CustomActions/SecurityHubCustomActions.yaml` CloudFormation Quindi modifica il `IRAutomation` ruolo in ciascuno degli account membro per consentire alla funzione Lambda che esegue l'azione di assumere il `IRAutomation` ruolo. Per ulteriori informazioni, consulta il GitHub repository.	

Risorse correlate

- AWS Guida alla risposta agli incidenti di sicurezza

Informazioni aggiuntive

Utilizzando questo ambiente, un team del Security Operations Center (SOC) può migliorare il processo di risposta agli incidenti di sicurezza attraverso quanto segue:

- Avere la capacità di eseguire analisi forensi in un ambiente separato per evitare la compromissione accidentale delle risorse di produzione
- Disporre di un processo standardizzato, ripetibile e automatizzato per il contenimento e l'analisi.
- Offrire a qualsiasi proprietario o amministratore di account la possibilità di avviare il processo di risposta agli incidenti con una conoscenza minima di come utilizzare i tag
- Disporre di un ambiente standardizzato e pulito per eseguire l'analisi degli incidenti e le indagini forensi senza il rumore di un ambiente più ampio
- Avere la capacità di creare più ambienti di analisi in parallelo

- Concentrare le risorse SOC sulla risposta agli incidenti anziché sulla manutenzione e la documentazione di un ambiente di analisi forense cloud
- Passare da un processo manuale a uno automatizzato per raggiungere la scalabilità
- Utilizzo CloudFormation di modelli per garantire la coerenza e per evitare attività ripetibili

Inoltre, eviti di utilizzare un'infrastruttura persistente e paghi le risorse quando ne hai bisogno.

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Controlla automaticamente i gruppi AWS di sicurezza che consentono l'accesso da indirizzi IP pubblici

Eugene Shifer e Stephen DiCato, Amazon Web Services

Riepilogo

Come best practice in materia di sicurezza, è fondamentale ridurre al minimo l'esposizione delle AWS risorse solo a ciò che è assolutamente necessario. Ad esempio, i server Web destinati al pubblico in generale devono consentire l'accesso in entrata da Internet, ma l'accesso ad altri carichi di lavoro deve essere limitato a reti specifiche per ridurre l'esposizione non necessaria. I gruppi di sicurezza in Amazon Virtual Private Cloud (Amazon VPC) sono un controllo efficace per aiutarti a limitare l'accesso alle risorse. Tuttavia, la valutazione dei gruppi di sicurezza può essere un compito complicato, specialmente nelle architetture con più account. AWS Config regole e AWS Security Hub CSPM controlli possono aiutarti a identificare i gruppi di sicurezza che consentono l'accesso dalla rete Internet pubblica (0.0.0.0/0) a protocolli di comunicazione di rete specifici, come Secure Shell (SSH), HTTP, HTTPS e Windows Remote Desktop Protocol (RDP). Tuttavia, queste regole e controlli non sono applicabili se i servizi vengono eseguiti su porte non standard o se l'accesso è limitato a determinati indirizzi IP pubblici. Ad esempio, ciò potrebbe verificarsi quando un servizio Web è associato alla porta TCP 8443 anziché alla porta TCP standard 443. Ciò può verificarsi anche quando gli sviluppatori hanno accesso al server dalle proprie reti domestiche, ad esempio a scopo di test.

Per risolvere questo problema, è possibile utilizzare la soluzione Infrastructure as Code (IaC) fornita in questo modello per identificare i gruppi di sicurezza che consentono l'accesso da qualsiasi indirizzo IP non privato (non conforme alla RFC 1918) a qualsiasi carico di lavoro dell'azienda o dell'organizzazione. Account AWS AWS Il AWS CloudFormationmodello fornisce una AWS Config regola personalizzata, una funzione e le autorizzazioni necessarie. AWS Lambda Puoi distribuirlo come stack in un singolo account o come stack impostato nell'intera organizzazione, gestito tramite. AWS Organizations

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Esperienza nell'utilizzo GitHub

- Se stai distribuendo in un'unica Account AWS soluzione:
  - Autorizzazioni per creare pile CloudFormation
  - AWS Config configurato nell'account di destinazione
  - (Facoltativo) Security Hub CSPM configurato nell'account di destinazione
- Se stai distribuendo in un'organizzazione: AWS
  - Autorizzazioni per creare set di stack CloudFormation
  - Configurazione CSPM di Security Hub con integrazione AWS Organizations
  - AWS Config configuralo negli account in cui stai distribuendo questa soluzione
  - Account AWS Designare un amministratore delegato per AWS Config Sand Security Hub CSPM

Limitazioni

- Se esegui la distribuzione su un account individuale che non ha il CSPM di Security Hub abilitato, puoi utilizzarlo AWS Config per valutare i risultati.
- Se stai eseguendo la distribuzione in un'organizzazione che non dispone di un amministratore delegato per AWS Config Security Hub CSPM, devi accedere agli account dei singoli membri per visualizzare i risultati.
- Se lo utilizzi AWS Control Tower per gestire e governare gli account della tua organizzazione, distribuisci IaC secondo questo schema utilizzando Customizations for (cFCT). AWS Control Tower L'utilizzo della CloudFormation console comporterebbe differenze di configurazione e richiederebbe la AWS Control Tower nuova registrazione delle unità organizzative () o degli account gestiti. OUs
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Architecture

Implementazione in un individuo Account AWS

Il seguente diagramma di architettura mostra l'implementazione delle AWS risorse all'interno di una singola unità. Account AWS Il provisioning delle risorse viene effettuato utilizzando un CloudFormation modello direttamente tramite la CloudFormation console. Se Security Hub CSPM è abilitato, è possibile visualizzare i risultati in uno AWS Config o in Security Hub CSPM. Se Security Hub CSPM non è abilitato, è possibile visualizzare i risultati solo in. AWS Config

Il diagramma mostra il flusso di lavoro seguente:

1. Crei uno stack CloudFormation . Questo implementa una funzione Lambda e AWS Config una regola. Sia la regola che la funzione sono configurate con le autorizzazioni AWS Identity and Access Management (IAM) necessarie per pubblicare le valutazioni delle risorse nei AWS Config log.
2. La AWS Config regola funziona in modalità di valutazione investigativa e richiama la funzione Lambda ogni 24 ore.
3. La funzione Lambda valuta i gruppi di sicurezza e invia aggiornamenti a. AWS Config
4. Security Hub CSPM riceve tutti i AWS Config risultati.
5. È possibile visualizzare i risultati in Security Hub CSPM o in AWS Config, a seconda dei servizi configurati nell'account.

Implementazione in un'organizzazione AWS

Il diagramma seguente mostra la distribuzione del pattern su più account gestiti tramite AWS Organizations e. AWS Control Tower Il CloudFormation modello viene distribuito tramite cFCT. I risultati della valutazione sono centralizzati in Security Hub CSPM nell'account amministratore delegato. La sezione del diagramma relativa al AWS CodePipeline flusso di lavoro mostra i passaggi di base che si verificano durante l'implementazione di cFCT.

Il diagramma mostra il flusso di lavoro seguente:

1. Nell'account di gestione, carichi un file compresso (ZIP) del modello IaC in un bucket Amazon Simple Storage Service (Amazon S3) distribuito da cFCT.
2. La pipeline cFct decomprime il file, esegue i controlli cfn-nag () e distribuisce il modello come set di stack. GitHub CloudFormation
3. A seconda della configurazione specificata nel file manifest cFCT, distribuisce gli stack in account individuali o specificati. CloudFormation StackSets OUs Questo implementa una funzione Lambda e AWS Config una regola negli account di destinazione. Sia la regola che la funzione sono configurate con le autorizzazioni IAM necessarie per pubblicare le valutazioni delle risorse nei AWS Config log.
4. La AWS Config regola funziona in modalità di valutazione investigativa e richiama la funzione Lambda ogni 24 ore.
5. La funzione Lambda valuta i gruppi di sicurezza e invia aggiornamenti a. AWS Config

6. AWS Config inoltra tutti i risultati a Security Hub CSPM.
7. I risultati del Security Hub CSPM vengono aggregati nell'account amministratore delegato.
8. È possibile visualizzare i risultati aggregati in Security Hub CSPM nell'account amministratore delegato.

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationti aiuta a configurare AWS le risorse, a fornirle in modo rapido e coerente e a gestirle durante tutto il loro ciclo di vita attraverso e. Account AWS Regioni AWS
- AWS Configfornisce una visione dettagliata delle risorse presenti Account AWS e di come sono configurate. Ti aiuta a identificare in che modo le risorse sono correlate tra loro e in che modo le loro configurazioni sono cambiate nel tempo. Una AWS Config regola definisce le impostazioni di configurazione ideali per una risorsa e consente di AWS Config valutare se le AWS risorse soddisfano le condizioni della regola.
- AWS Control Towerti aiuta a configurare e gestire un ambiente con AWS più account, seguendo le migliori pratiche prescrittive. Customizations for AWS Control Tower (cFct) ti aiuta a personalizzare la tua AWS Control Tower landing zone e a rimanere in linea con AWS le migliori pratiche. Le personalizzazioni di questa soluzione sono implementate tramite CloudFormation modelli e politiche di controllo del AWS Organizations servizio (). SCPs
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS Organizationsè un servizio di gestione degli account che ti aiuta a consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente.
- AWS Security Hub CSPMfornisce una visione completa dello stato di sicurezza in AWS. Inoltre, consente di verificare la conformità AWS dell'ambiente agli standard e alle best practice del settore della sicurezza.

Altri strumenti

- Python è un linguaggio di programmazione per computer generico.

Archivio di codice

Il codice di questo modello è disponibile nell'archivio GitHub Detect vulnerable security groups.

Best practice

Ti consigliamo di attenerti alle migliori pratiche riportate nelle seguenti risorse:

- Best practice per le unità organizzative con AWS Organizations (blog AWS Cloud Operations & Migrations)
- Guida per la creazione di una base iniziale utilizzando AWS Control Tower on AWS (AWS Solutions Library)
- Linee guida per la creazione e la modifica AWS Control Tower delle risorse (AWS Control Tower documentazione)
- Considerazioni sull'implementazione del cFCT (documentazione)AWS Control Tower
- Applica le autorizzazioni con privilegi minimi (documentazione IAM)

Epiche

Rivedi il modello CloudFormation

Operazione	Description	Competenze richieste
Determina la tua strategia di implementazione.	Esamina la soluzione e il codice per determinare la strategia di implementazione per il tuo AWS ambiente. Determina se stai implement ando in un unico account o in un' AWS organizzazione.	Proprietario dell'app, General AWS
Clonare il repository.	Inserisci il seguente comando per clonare il repository Detect vulnerable security groups: `git clone https://github.com/aws-samples/detect-public-security-groups.git`	Sviluppatore di app, proprietario dell'app

Operazione	Description	Competenze richieste
Convalida la versione di Python.	1. Naviga nella directory di primo livello nel repository clonato: `cd detect-public-security-groups` 2. Apri Security-Group-Public-Assessment.yaml. 3. Nella `SgPublicAccessCheckLambdaFunction` risorsa, verifica che la versione di Python sia compatibile con il tuo obiettivo. Regioni AWS Per impostazione predefinita, questa funzione utilizza Python 3.12. Per ulteriori informazioni, consulta AWS Lambda aggiunge il supporto per Python 3.12. Se necessario, aggiorna la versione di Python. 4. Salvate e chiudete Security-Group-Public-Assessment.yaml.	Amministratore AWS, sviluppatore di app

Implementa il modello CloudFormation

Operazione	Description	Competenze richieste
Implementa il CloudFormation modello.	Implementa il CloudFormation modello nel tuo AWS	Sviluppatore di app, amministratore AWS, General AWS

Operazione	Description	Competenze richieste
	ambiente. Esegui una delle seguenti operazioni: • Se stai eseguendo la distribuzione su un singolo dispositivo Account AWS, segui le istruzioni riportate in Creazione di uno stack. • Se stai eseguendo la distribuzione in un'organizzazione che non è gestita da AWS Control Tower, segui le istruzioni in Creare un set di stack. • Se esegui la distribuzione in un'organizzazione gestita da AWS Control Tower, consulta le istruzioni nella sezione Crea le tue personalizzazioni.	
Verifica la distribuzione.	Nella CloudFormation console, verifica che lo stack o il set di stack siano stati distribuiti correttamente.	Amministratore AWS, proprietario dell'app

Esamina i risultati

Operazione	Description	Competenze richieste
Visualizza i risultati delle AWS Config regole.	In Security Hub CSPM, procedi come segue per visualizzare un elenco di risultati individuali:	Amministratore AWS, amministratore di sistema AWS, amministratore cloud

Operazione	Description	Competenze richieste
	1. Apri la console Security Hub CSPM. 2. Nel riquadro di navigazione, seleziona Esiti. 3. Nella casella Aggiungi filtri, aggiungi i seguenti filtri: • Lo stato di conformità è `FAILED` • Il titolo è `SgPublicAccessCheck` 4. Scegli Applica. In Security Hub CSPM, procedi come segue per visualizzare un elenco di risultati totali raggruppati per: Account AWS 1. Apri la console Security Hub CSPM. 2. Nel riquadro di navigazione, seleziona Informazioni dettagliate. 3. Scegliere Create insight (Crea informazione dettagliata). 4. Per selezionare l'attributo di raggruppamento per l'informazione dettagliata: a. Scegli la casella di ricerca per visualizzare le opzioni di filtro.	

Operazione	Description	Competenze richieste
	b. Scegliere Group by (Raggruppa per). c. Seleziona AwsAccountId. d. Scegli Applica. 5. Nella casella Aggiungi filtri, aggiungi i seguenti filtri: • Il titolo è `SgPublicAccessCheck` • Lo stato di conformità è `FAILED` 6. Scegliere Create insight (Crea informazione dettagliata). 7. Inserisci un nome per Insight, quindi scegli Create insight. Inoltre AWS Config, per visualizzare un elenco di risultati, segui le istruzioni in Visualizzazione delle informazioni sulla conformità e dei risultati della valutazione nella AWS Config documentazione.	

risoluzione dei problemi

Problema	Soluzione
La creazione o l'eliminazione CloudFormation dello stack set non riesce.	Quando AWS Control Tower viene distribuito, impone i guardrail necessari e assume il controllo degli aggregatori e delle regole.

Problema	Soluzione
	AWS Config Ciò include la prevenzione di eventuali modifiche dirette. CloudFormation Per distribuire o rimuovere correttamente questo CloudFormation modello, incluse tutte le risorse associate, è necessario utilizzare cFCT.
cFct non riesce a eliminare il modello. CloudFormation	Se il CloudFormation modello persiste anche dopo aver apportato le modifiche necessarie nel file manifesto e aver rimosso i file modello, verificate che il file manifesto contenga il `enable_stack_set_deletion` parametro e che il valore sia impostato su. `false` Per ulteriori informazioni, consulta Eliminare uno stack set nella documentazione cFct.

Risorse correlate

- AWS Config Regole personalizzate (documentazione)AWS Config

Automatically remediate unencrypted Amazon RDS DB instances and clusters

Ajay Rawat e Josh Joy, Amazon Web Services

Riepilogo

Questo modello descrive come correggere automaticamente le istanze e i cluster di database di Amazon Relational Database Service (Amazon RDS) non crittografati su Amazon Web Services AWS() AWS Config utilizzando AWS Systems Manager , runbook e () chiavi. AWS Key Management Service AWS KMS

Le istanze DB RDS crittografate forniscono un ulteriore livello di protezione dei dati proteggendo i dati dall'accesso non autorizzato allo storage sottostante. Puoi utilizzare la crittografia Amazon RDS per aumentare la protezione dei dati delle tue applicazioni distribuite in e per soddisfare i requisiti di conformità per la crittografia a riposo. Cloud AWS Puoi abilitare la crittografia per un'istanza DB RDS al momento della creazione, ma non dopo la creazione. Tuttavia, è possibile aggiungere la crittografia a un'istanza RDS DB non crittografata creando uno snapshot dell'istanza DB e quindi creando una copia crittografata di tale istantanea. È quindi possibile ripristinare un'istanza DB dallo snapshot crittografato per ottenere una copia crittografata dell'istanza DB originale.

Questo modello viene utilizzato Regole di AWS Config per valutare le istanze e i cluster DB RDS. Applica la correzione utilizzando AWS Systems Manager i runbook, che definiscono le azioni da eseguire su risorse Amazon RDS non conformi, AWS KMS e le chiavi per crittografare gli snapshot DB. Quindi applica le politiche di controllo del servizio (SCPs) per impedire la creazione di nuove istanze e cluster DB senza crittografia.

Il codice per questo modello è fornito in. GitHub

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- File dal repository del codice GitHub sorgente relativo a questo pattern scaricati sul computer
- Un'istanza o un cluster RDS DB non crittografato
- Una AWS KMS chiave esistente per la crittografia di istanze e cluster RDS DB
- Accesso per aggiornare la politica delle risorse chiave del KMS

- AWS Config abilitato nel tuo Account AWS (vedi Guida introduttiva AWS Config nella AWS documentazione)

Limitazioni

- È possibile abilitare la crittografia per un'istanza DB RDS solo al momento della creazione, non dopo la creazione.
- Non è possibile creare una replica di lettura crittografata di un'istanza database non crittografata o una replica di lettura non crittografata di un'istanza database crittografata.
- Non puoi ripristinare un backup o uno snapshot non crittografato in un'istanza database crittografata.
- La crittografia Amazon RDS è disponibile per la maggior parte delle classi di istanza database. Per un elenco di eccezioni, consulta Encrypting Amazon RDS resources nella documentazione di Amazon RDS.
- Per copiare uno snapshot crittografato da uno Regione AWS all'altro, devi specificare la chiave KMS nella destinazione. Regione AWS Questo perché le chiavi KMS sono specifiche del tipo in Regione AWS cui vengono create.
- La snapshot di origine resta crittografata nel processo di copia. Amazon RDS utilizza la crittografia a busta per proteggere i dati durante il processo di copia. Per ulteriori informazioni, consulta Envelope encryption nella documentazione. AWS KMS
- Non è possibile decrittografare un'istanza DB crittografata. Tuttavia, è possibile esportare dati da un'istanza DB crittografata e importarli in un'istanza DB non crittografata.
- Dovresti eliminare una chiave KMS solo quando sei sicuro di non averne più bisogno. Se non sei sicuro, prendi in considerazione la possibilità di disabilitare la chiave KMS anziché eliminarla. Puoi riattivare una chiave KMS disabilitata se devi riutilizzarla in un secondo momento, ma non puoi recuperare una chiave KMS eliminata.
- Se scegli di non conservare i backup automatici, i backup automatici che si trovano nella stessa istanza DB vengono Regione AWS eliminati. Non potranno quindi essere recuperati dopo aver eliminato l'istanza database.
- I backup automatici vengono conservati per il periodo di conservazione impostato sull'istanza DB al momento dell'eliminazione. Questo periodo di conservazione impostato si verifica se si sceglie o meno di creare uno snapshot DB finale.
- Se la riparazione automatica è abilitata, questa soluzione crittografa tutti i database che hanno la stessa chiave KMS.

Architecture

Il diagramma seguente illustra l'architettura per l'implementazione. CloudFormation Si noti che è possibile implementare questo modello anche utilizzando. AWS Cloud Development Kit (AWS CDK)

Tools (Strumenti)

Strumenti

- CloudFormationti aiuta a configurare automaticamente le tue risorse. AWS Consente di utilizzare un file modello per creare ed eliminare una raccolta di risorse insieme come una singola unità (una pila).
- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software per definire l'infrastruttura cloud in codice e fornirla utilizzando linguaggi di programmazione familiari.

Servizi e funzionalità AWS

- AWS Configtiene traccia della configurazione delle tue AWS risorse e delle loro relazioni con le altre risorse. Può anche valutare la conformità di tali AWS risorse. Questo servizio utilizza regole che possono essere configurate per valutare AWS le risorse rispetto alle configurazioni desiderate. È possibile utilizzare un set di regole AWS Config gestite per scenari di conformità comuni oppure creare regole personalizzate per scenari personalizzati. Quando una AWS risorsa risulta non conforme, puoi specificare un'azione di riparazione tramite un AWS Systems Manager runbook e, facoltativamente, inviare un avviso tramite un argomento Amazon Simple Notification Service (Amazon SNS). In altre parole, puoi associare le azioni di riparazione Regole di AWS Config e scegliere di eseguirle automaticamente per risolvere il problema delle risorse non conformi senza intervento manuale. Se una risorsa non è ancora conforme dopo la riparazione automatica, è possibile impostare la regola per riprovare la riparazione automatica.
- Amazon Relational Database Service (Amazon RDS) semplifica la configurazione, il funzionamento e la scalabilità di un database relazionale nel cloud. L'elemento costitutivo di base di Amazon RDS è l'istanza DB, che è un ambiente di database isolato in. Cloud AWS Amazon RDS offre una selezione di tipi di istanze ottimizzati per adattarsi a diversi casi d'uso di database relazionali. I tipi di istanza comprendono varie combinazioni di CPU, memoria, storage e capacità di rete e offrono la flessibilità necessaria per scegliere la combinazione di risorse appropriata per il database. Ogni tipo di istanza include diverse dimensioni di istanza, che consentono di scalare il database in base ai requisiti del carico di lavoro di destinazione.

- AWS Key Management Service (AWS KMS) è un servizio gestito che semplifica la creazione e il controllo AWS KMS keys e crittografa i dati. Una chiave KMS è una rappresentazione logica di una chiave radice. La chiave KMS include metadati, ad esempio l'ID della chiave, la data di creazione, la descrizione e lo stato della chiave.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- Le policy di controllo dei servizi (SCPs) offrono il controllo centralizzato sulle autorizzazioni massime disponibili per tutti gli account dell'organizzazione. SCPs ti aiutano a garantire che i tuoi account rispettino le linee guida per il controllo degli accessi della tua organizzazione. SCPs non influiscono sugli utenti o sui ruoli nell'account di gestione. Influiscono solo sugli account membri nell'organizzazione. Ti consigliamo vivamente di non collegarti SCPs alla radice della tua organizzazione senza aver testato a fondo l'impatto che la politica ha sugli account. Crea invece un'unità organizzativa (OU) in cui trasferire i tuoi account uno alla volta, o almeno in piccoli numeri, per assicurarti di non bloccare inavvertitamente gli utenti dall'accesso ai servizi chiave.

Codice

Il codice sorgente e i modelli di questo pattern sono disponibili in un GitHub repository. Il modello offre due opzioni di implementazione: è possibile distribuire un CloudFormation modello per creare il ruolo di riparazione che crittografa le istanze e i cluster DB RDS oppure utilizzare il. AWS CDK Il repository ha cartelle separate per queste due opzioni.

La sezione Epics fornisce step-by-step istruzioni per la distribuzione del modello. CloudFormation Se desideri utilizzare il AWS CDK, segui le istruzioni contenute nel `README.md` file nel GitHub repository.

Best practice

- Abilita la crittografia dei dati sia a riposo che in transito.
- Abilita AWS Config in tutti gli account e Regioni AWS.
- Registra le modifiche alla configurazione di tutti i tipi di risorse.
- Ruota periodicamente le credenziali IAM.
- Sfrutta l'etichettatura per AWS Config, che semplifica la gestione, la ricerca e il filtraggio delle risorse.

Epiche

Crea il ruolo di riparazione IAM e il runbook Systems Manager

Operazione	Description	Competenze richieste
Scarica il CloudFormation modello.	Scarica il `unencrypted-to-encrypted-rds.template.json` file dal GitHub repository.	DevOps ingegnere
Crea lo CloudFormation stack.	1. Accedi a Console di gestione AWS e apri la CloudFormation console. 2. Avvia il `unencrypted-to-encrypted-rds.template.json` modello per creare un nuovo stack. Per ulteriori informazioni sulla distribuzione dei modelli, consulta la CloudFormation documentazione.	DevOps ingegnere
Rivedi CloudFormation parametri e valori.	1. Rivedi i dettagli dello stack e aggiorna i valori in base ai requisiti dell'ambiente. 2. Scegli Crea stack per distribuire il modello.	DevOps ingegnere
Rivedi le risorse.	Quando lo stack è stato creato, il suo stato cambia in CREATE_COMPLETE. Esamina le risorse create (ruolo IAM, runbook Systems	DevOps ingegnere

Operazione	Description	Competenze richieste
	Manager) nella CloudForm ation console.	

Aggiorna la politica AWS KMS chiave

Operazione	Description	Competenze richieste
Aggiorna la tua politica sulle chiavi KMS.	1. Assicurati che l'alias `alias/RDSEncryptio nAtRestKMSAlias` della chiave esista. 2. La dichiarazione politica chiave dovrebbe includere il ruolo di riparazione IAM. (Controlla le risorse create dal CloudFormation modello che hai distribuito nell'epop ea precedente.) 3. Nella seguente politica chiave, aggiorna le parti in grassetto in modo che corrispondano al tuo account e al ruolo IAM che è stato creato. `{ "Sid": "Allow access through RDS for all principals in the account that are authorized to use RDS", "Effect": "Allow", "Principal": {`	DevOps ingegnere

Operazione	Description	Competenze richieste
	"AWS": "arn:aws:iam:: <your-AWS-account-ID>":role/<your-IAM-remediation-role>" }, "Action": ["kms:Encrypt", "kms:Decrypt", "kms:ReEncrypt*", "kms:GenerateDataKey*", "kms:CreateGrant", "kms:ListGrants", "kms:DescribeKey"], "Resource": "*", "Condition": { "StringEquals": { "kms:ViaService": "rds.us-east-1.amazonaws.com", "kms:CallerAccount": "<your-AWS-account-ID>" } } }	

Trova e correggi le risorse non conformi

Operazione	Description	Competenze richieste
Visualizza le risorse non conformi.	1. Per visualizzare un elenco di risorse non conformi, apri la console.AWS Config 2. Nel riquadro di navigazione, scegli Regole, quindi scegli la regola. `rds-storage-encrypted` Le risorse non conformi elencate nella AWS Config console saranno istanze, non cluster. L'automazione della riparazione crittografa le istanze e i cluster e crea una nuova istanza crittografata o un cluster appena creato. Tuttavia, assicurati di non correggere contemporaneamente più istanze che appartengono allo stesso cluster. Prima di correggere eventuali istanze o volumi DB RDS, assicurati che l'istanza DB RDS non sia in uso. Verifica che non siano in corso operazioni di scrittura durante la creazione dello snapshot, per assicurarti che l'istantanea contenga i dati originali. Valuta la possibilità di applicare	DevOps ingegnere

Operazione	Description	Competenze richieste
	una finestra di manutenzione durante la quale verrà eseguita la riparazione.	
Correggi le risorse non conformi.	1. Quando sei pronto e la finestra di manutenzione è attiva, scegli la risorsa da correggere, quindi scegli Ripara. La colonna Stato dell'azione dovrebbe ora mostrare l'Azione in coda di esecuzione. 2. Visualizza l'avanzamento e lo stato della riparazione in Systems Manager. Aprire la console Systems Manager. Nel riquadro di navigazione, scegli Automazione, quindi seleziona l'ID di esecuzione dell'automazione corrispondente per visualizzare ulteriori dettagli.	DevOps ingegnere

Operazione	Description	Competenze richieste
Verifica che l'istanza DB RDS sia disponibile.	Una volta completata l'automazione, la nuova istanza DB RDS crittografata sarà disponibile. L'istanza DB RDS crittografata avrà il prefisso `encrypted` seguito dal nome originale . Ad esempio, se il nome dell'istanza DB RDS non crittografata fosse`database-1` , lo sarebbe l'istanza DB RDS appena crittografata. `encrypted-database-1`	DevOps ingegnere
Termina l'istanza non crittografata.	Una volta completata la riparazione e convalidata la nuova risorsa crittografata, è possibile terminare l'istanza non crittografata. Assicurati di confermare che la nuova risorsa crittografata corrisponda alla risorsa non crittografata prima di terminare qualsiasi risorsa.	DevOps ingegnere

Far rispettare SCPs

Operazione	Description	Competenze richieste
Far rispettare SCPs.	SCPs Implementa per impedire che in futuro vengano creati istanze e cluster di database senza crittografia. Utilizza il	Ingegnere della sicurezza

Operazione	Description	Competenze richieste
	`rds_encrypted.json` file fornito nel GitHub repository per questo scopo e segui le istruzioni contenute nella documentazione.AWS	

Risorse correlate

Riferimenti

- Configurazione AWS Config
- AWS Config regole personalizzate
- AWS KMS concetti
- AWS Systems Manager documenti
- Policy di controllo dei servizi

Strumenti

- CloudFormation
- AWS Cloud Development Kit (AWS CDK)

Guide e pattern

- Riattiva automaticamente AWS CloudTrail utilizzando una regola di riparazione personalizzata in AWS Config

Informazioni aggiuntive

Come funziona AWS Config ?

Quando lo usi AWS Config, rileva innanzitutto AWS le risorse supportate presenti nel tuo account e genera un elemento di configurazione per ogni risorsa. AWS Config genera inoltre elementi di configurazione quando la configurazione di una risorsa cambia e conserva i record cronologici degli elementi di configurazione delle risorse dal momento in cui si avvia il registratore di configurazione.

Per impostazione predefinita, AWS Config crea elementi di configurazione per ogni risorsa supportata in. Regione AWS Se non si desidera AWS Config creare elementi di configurazione per tutte le risorse supportate, è possibile specificare i tipi di risorse di cui si desidera che venga monitorato.

In che modo sono AWS Config e sono Regole di AWS Config correlati a AWS Security Hub CSPM?

AWS Security Hub CSPM è un servizio di sicurezza e conformità che fornisce la gestione della situazione di sicurezza e conformità come servizio. Utilizza AWS Config e Regole di AWS Config come meccanismo principale per valutare la configurazione delle AWS risorse. Regole di AWS Config può essere utilizzato anche per valutare direttamente la configurazione delle risorse. Altro Servizi AWS, tale AWS Control Tower e AWS Firewall Manager altro uso Regole di AWS Config.

Convalida e distribuisci automaticamente le policy e i ruoli IAM utilizzando CodePipeline IAM Access Analyzer e le macro AWS CloudFormation

Helton Ribeiro e Guilherme Simoes, Amazon Web Services

Riepilogo

Questo modello descrive i passaggi e fornisce il codice per creare una pipeline di distribuzione che consenta ai team di sviluppo di creare policy e ruoli AWS Identity and Access Management (IAM) nei tuoi account Amazon Web Services (AWS). Questo approccio aiuta l'organizzazione a ridurre il sovraccarico per i team operativi e ad accelerare il processo di implementazione. Inoltre, aiuta gli sviluppatori a creare ruoli e policy IAM compatibili con i controlli di governance e sicurezza esistenti.

L'approccio di questo modello utilizza AWS Identity and Access Management Access Analyzer per convalidare le policy IAM da collegare ai ruoli IAM e utilizza AWS CloudFormation per distribuire i ruoli IAM. Tuttavia, invece di modificare direttamente il file CloudFormation modello AWS, il tuo team di sviluppo crea policy e ruoli IAM in formato JSON. Una CloudFormation macro AWS trasforma questi file di policy in formato JSON in tipi di risorse AWS CloudFormation IAM prima di iniziare la distribuzione.

La pipeline di distribuzione (`RolesPipeline`) ha fasi di origine, convalida e distribuzione. Durante la fase di origine, il tuo team di sviluppo invia i file JSON che contengono la definizione dei ruoli e delle policy IAM a un repository CodeCommit AWS. AWS esegue CodeBuild quindi uno script per convalidare tali file e li copia in un bucket Amazon Simple Storage Service (Amazon S3). Poiché i tuoi team di sviluppo non hanno accesso diretto al file CloudFormation modello AWS archiviato in un bucket S3 separato, devono seguire il processo di creazione e convalida del file JSON.

Infine, durante la fase di implementazione, AWS CodeDeploy utilizza uno CloudFormation stack AWS per aggiornare o eliminare le policy e i ruoli IAM in un account.

> **Important**
>
> Il flusso di lavoro di questo pattern è un proof of concept (POC) e ti consigliamo di utilizzarlo solo in un ambiente di test. Se desideri utilizzare l'approccio di questo modello in un ambiente di produzione, consulta le migliori pratiche di sicurezza in IAM nella documentazione IAM e apporta le modifiche necessarie ai tuoi ruoli IAM e ai servizi AWS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un bucket S3 nuovo o esistente per la `RolesPipeline` pipeline. Assicurati che le credenziali di accesso che stai utilizzando dispongano delle autorizzazioni per caricare oggetti in questo bucket.
- AWS Command Line Interface (AWS CLI), installata e configurata. Per ulteriori informazioni su questo argomento, consulta Installazione, aggiornamento e disinstallazione dell'interfaccia a riga di comando di AWS nella documentazione dell'interfaccia a riga di comando di AWS.
- AWS Serverless Application Model (AWS SAM) Serverless Application Model (AWS) Cli, installata e configurata. Per ulteriori informazioni su questo argomento, consulta Installazione della CLI AWS SAM nella documentazione di AWS SAM.
- Python 3, installato sul computer locale. Per ulteriori informazioni su questo argomento, consulta la documentazione di Python.
- Un client Git, installato e configurato.
- Il GitHub `IAM roles pipeline` repository, clonato sul computer locale.
- Politiche e ruoli IAM esistenti in formato JSON. Per ulteriori informazioni su questo argomento, consulta il ReadMefile nel repository Github. `IAM roles pipeline`
- Il tuo team di sviluppatori non deve disporre delle autorizzazioni per modificare AWS CodePipeline e CodeDeploy le risorse di questa soluzione. CodeBuild

Limitazioni

- Il flusso di lavoro di questo pattern è un proof of concept (POC) e ti consigliamo di utilizzarlo solo in un ambiente di test. Se desideri utilizzare l'approccio di questo modello in un ambiente di produzione, consulta le migliori pratiche di sicurezza in IAM nella documentazione IAM e apporta le modifiche necessarie ai tuoi ruoli IAM e ai servizi AWS.

Architecture

Il diagramma seguente mostra come convalidare e distribuire automaticamente i ruoli e le policy IAM su un account utilizzando IAM Access CodePipeline Analyzer e le macro AWS. CloudFormation

Il diagramma mostra il flusso di lavoro seguente:

1. Uno sviluppatore scrive file JSON che contengono le definizioni per le politiche e i ruoli IAM. Lo sviluppatore inserisce il codice in un CodeCommit repository e CodePipeline quindi avvia la pipeline. `RolesPipeline`
2. CodeBuild convalida i file JSON utilizzando IAM Access Analyzer. Se vengono rilevati risultati relativi alla sicurezza o agli errori, il processo di distribuzione viene interrotto.
3. Se non ci sono risultati relativi alla sicurezza o agli errori, i file JSON vengono inviati al bucket S3. `RolesBucket`
4. Una CloudFormation macro AWS implementata come funzione AWS Lambda legge quindi i file JSON dal `RolesBucket` bucket e li trasforma in tipi di risorse AWS IAM. CloudFormation
5. Uno CloudFormation stack AWS predefinito installa, aggiorna o elimina le policy e i ruoli IAM nell'account.

Automazione e scalabilità

CloudFormation I modelli AWS che implementano automaticamente questo modello sono forniti nel repository di pipeline dei ruoli GitHub IAM.

Tools (Strumenti)

- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- IAM Access Analyzer ti aiuta a identificare le risorse della tua organizzazione e dei tuoi account, come i bucket S3 o i ruoli IAM, che sono condivisi con un'entità esterna. Questo ti aiuta a identificare gli accessi involontari alle tue risorse e ai tuoi dati.
- AWS Serverless Application Model (AWS SAM) Serverless Application Model (AWS SAM) è un framework open source che ti aiuta a creare applicazioni serverless nel cloud AWS.

Codice

Il codice sorgente e i modelli di questo modello sono disponibili nell'archivio della pipeline di ruoli GitHub IAM.

Epiche

Clona il repository

Operazione	Description	Competenze richieste
Clonare il repository di esempio.	Clona il repository della pipeline dei ruoli GitHub IAM sul tuo computer locale.	Sviluppatore di app, General AWS

Implementa la pipeline RolesPipeline

Operazione	Description	Competenze richieste
Implementa la pipeline.	1. Passa alla directory che contiene il repository clonato. 2. **Important** Esegui il `make deploy bucket=<bucket_name>` comando.: devi sostituire `<bucket_name>` con il nome del bucket S3 esistente. 3. Esegui il `aws codepipeline get-pipeline --name RolesPipeline` comando per verificare se la distribuzione è riuscita.	Sviluppatore di app, General AWS

Operazione	Description	Competenze richieste
Clona il repository della pipeline.	1. Lo CloudFormation stack `RolesPipeline` AWS crea il `roles-pipeline-repo` CodeCommit repository. 2. Accedi alla Console di gestione AWS, apri la CodeCommit console AWS e copia l'URL del CodeCommit repository per clonarlo sul tuo computer locale. Per ulteriori informazioni su questo argomento, consulta Connect to an AWS CodeCommit repository nella CodeCommit documentazione AWS.	Sviluppatore di app, General AWS

Testa la RolesPipeline pipeline

Operazione	Description	Competenze richieste
Testa la RolesPipeline pipeline con policy e ruoli IAM validi.	1. Crea file JSON per le tue politiche e i tuoi ruoli IAM. Puoi utilizzare gli esempi presenti nella `role-example` directory del GitHub `IAM roles pipeline` repository. 2. **Important** Definisci le politiche e i ruoli IAM con	Sviluppatore di app, General AWS

Operazione	Description	Competenze richieste
	le configurazioni richieste. : Assicurati di seguire il formato descritto nel `ReadMe` file del GitHub `IAM roles pipeline` repository. 3. Inserisci le modifiche nel repository. `roles-pipeline-repo` CodeCommit 4. Verifica l'implementazione della pipeline. `RolesPipeline` 5. Assicurati che le policy e i ruoli IAM siano implementati correttamente nell'account. 6. Verifica se esiste un limite di autorizzazioni associato alle politiche o ai ruoli IAM. Per ulteriori informazioni a riguardo, consulta i limiti delle autorizzazioni per le entità IAM nella documentazione IAM.	

Operazione	Description	Competenze richieste
Testa la RolesPipeline pipeline con policy e ruoli IAM non validi.	1. Modifica il `roles-pipeline-repo` CodeCommit repository e includi ruoli o policy IAM non validi. Ad esempio, puoi utilizzare un'azione che non esiste o una versione della policy IAM non valida. 2. Verifica l'implementazione della pipeline. IAM Access Analyzer interrompe la pipeline durante la fase di convalida se rileva policy o ruoli IAM non validi.	Sviluppatore di app, General AWS

Pulizia delle risorse

Operazione	Description	Competenze richieste
Preparati per la pulizia.	Svuota i bucket S3 e poi esegui il comando. `destroy`	Sviluppatore di app, General AWS
Elimina lo RolesStack stack.	1. La `RolesPipeline` pipeline crea uno CloudFormation stack `RolesStack` AWS che implementa le policy e i ruoli IAM. È necessari o eliminare questo stack prima di eliminare la pipeline. `RolesPipeline` 2. Accedi alla Console di gestione AWS, apri la CloudFormation console	Sviluppatore di app, General AWS

Operazione	Description	Competenze richieste
	AWS, quindi scegli lo `RolesStack` stack e scegli Elimina.	
Elimina lo RolesPipeline stack.	Per eliminare lo CloudFormation stack `RolesPipeline` AWS, segui le istruzioni dal ReadMefile nel repository Github`IAM roles pipeline`.	Sviluppatore di app, General AWS

Risorse correlate

- IAM Access Analyzer - Convalida delle policy (AWS News Blog)
- Utilizzo di CloudFormation macro AWS per eseguire elaborazioni personalizzate su modelli (CloudFormation documentazione AWS)
- Creazione di funzioni Lambda con Python (documentazione AWS Lambda)

Integrazione bidirezionale AWS Security Hub CSPM con il software Jira

Joaquín Rinaudo, Amazon Web Services

Riepilogo

Questa soluzione supporta un'integrazione bidirezionale tra e Jira. AWS Security Hub CSPM Utilizzando questa soluzione, puoi creare e aggiornare automaticamente e manualmente i ticket Jira dai risultati del Security Hub CSPM. I team addetti alla sicurezza possono utilizzare questa integrazione per notificare ai team di sviluppatori gravi problemi di sicurezza che richiedono un intervento.

La soluzione consente di:

- Seleziona i controlli CSPM di Security Hub per creare o aggiornare automaticamente i ticket in Jira.
- Nella console CSPM di Security Hub, usa le azioni personalizzate di Security Hub CSPM per inoltrare manualmente i ticket in Jira.
- Assegna automaticamente i ticket in Jira in base ai tag definiti in. Account AWS AWS Organizations Se questo tag non è definito, viene utilizzato un assegnatario predefinito.
- Elimina automaticamente i risultati CSPM di Security Hub contrassegnati come falsi positivi o rischi accettati in Jira.
- Chiudi automaticamente un ticket Jira quando i relativi risultati vengono archiviati in Security Hub CSPM.
- Riapri i ticket Jira quando si ripresentano i risultati CSPM di Security Hub.

Flusso di lavoro Jira

La soluzione utilizza un flusso di lavoro Jira personalizzato che consente agli sviluppatori di gestire e documentare i rischi. Man mano che il problema passa attraverso il flusso di lavoro, l'integrazione bidirezionale assicura che lo stato del ticket Jira e del risultato CSPM di Security Hub sia sincronizzato tra i flussi di lavoro di entrambi i servizi. Questo flusso di lavoro è un derivato di SecDevOps Risk Workflow di Dinis Cruz, concesso in licenza con licenza Apache License versione 2.0. Ti consigliamo di aggiungere una condizione del flusso di lavoro di Jira in modo che solo i membri del team di sicurezza possano modificare lo stato del ticket.

Per un esempio di ticket Jira generato automaticamente da questa soluzione, consulta la sezione Informazioni aggiuntive di questo modello.

Prerequisiti e limitazioni

Prerequisiti

- Se desideri implementare questa soluzione in un ambiente con più AWS account:
  - Il tuo ambiente multi-account è attivo e gestito da. AWS Organizations
  - Security Hub CSPM è abilitato sul tuo. Account AWS
  - Nel AWS Organizations, hai designato un account amministratore CSPM di Security Hub.
  - Hai un ruolo multiaccount AWS Identity and Access Management (IAM) che dispone delle `AWSOrganizationsReadOnlyAccess` autorizzazioni per l'account di gestione. AWS Organizations
  - (Facoltativo) Hai taggato il tuo Account AWS con. `SecurityContactID` Questo tag viene utilizzato per assegnare i ticket Jira ai contatti di sicurezza definiti.
- Se desideri implementare questa soluzione in un'unica soluzione: Account AWS
  - Hai un attivo Account AWS.
  - Security Hub CSPM è abilitato sul tuo. Account AWS
- Un'istanza di Jira Data Center

> **Important**
>
> Questa soluzione supporta l'uso di Jira Cloud. Tuttavia, Jira Cloud non supporta l'importazione di flussi di lavoro XML, quindi è necessario ricreare manualmente il flusso di lavoro in Jira. Puoi trovare le transizioni e lo stato nel repository. GitHub

- Autorizzazioni di amministratore in Jira
- Uno dei seguenti token Jira:
  - Per Jira Enterprise, un token di accesso personale (PAT). Per ulteriori informazioni, consulta Utilizzo dei token di accesso personali (supporto Atlassian).
  - Per Jira Cloud, un token API Jira. Per ulteriori informazioni, consulta Gestire i token API (supporto Atlassian).

Architecture

Questa sezione illustra l'architettura della soluzione in vari scenari, ad esempio quando lo sviluppatore e il tecnico della sicurezza decidono di accettare il rischio o decidono di risolvere il problema.

Scenario 1: lo sviluppatore risolve il problema

1. Security Hub CSPM genera un risultato rispetto a un controllo di sicurezza specifico, come quelli dello standard AWS Foundational Security Best Practices.
2. Un CloudWatch evento Amazon associato alla scoperta e all'`CreateJIRAa`zione avvia una AWS Lambda funzione.
3. La funzione Lambda utilizza il proprio file di configurazione e il `GeneratorId` campo del risultato per valutare se è necessario aumentare la scalabilità del risultato.
4. La funzione Lambda determina che il risultato deve essere incrementato e ottiene il tag dell'account dall'`SecurityContactID` AWS Organizations account di gestione. AWS Questo ID è associato allo sviluppatore e viene utilizzato come ID assegnatario per il ticket Jira.
5. La funzione Lambda utilizza le credenziali archiviate in Gestione dei segreti AWS per creare un ticket in Jira. Jira avvisa lo sviluppatore.
6. Lo sviluppatore risolve il problema di sicurezza sottostante e, in Jira, modifica lo stato del ticket in. `TEST FIX`
7. Security Hub CSPM aggiorna i `ARCHIVED` risultati e viene generato un nuovo evento. Questo evento fa sì che la funzione Lambda chiuda automaticamente il ticket Jira.

Scenario 2: lo sviluppatore decide di accettare il rischio

1. Security Hub CSPM genera un risultato rispetto a un controllo di sicurezza specifico, come quelli dello standard AWS Foundational Security Best Practices.
2. Un CloudWatch evento associato al risultato e all'`CreateJIRAa`zione avvia una funzione Lambda.
3. La funzione Lambda utilizza il proprio file di configurazione e il `GeneratorId` campo del risultato per valutare se è necessario aumentare la scalabilità del risultato.
4. La funzione Lambda determina che il risultato deve essere incrementato e ottiene il tag dell'account dall'`SecurityContactID` AWS Organizations account di gestione. AWS Questo ID è associato allo sviluppatore e viene utilizzato come ID assegnatario per il ticket Jira.

5. La funzione Lambda utilizza le credenziali archiviate in Secrets Manager per creare un ticket in Jira. Jira avvisa lo sviluppatore.
6. Lo sviluppatore decide di accettare il rischio e, in Jira, modifica lo stato del ticket in. `AWAITING RISK ACCEPTANCE`
7. L'ingegnere della sicurezza esamina la richiesta e ritiene che la giustificazione aziendale sia appropriata. L'ingegnere della sicurezza modifica lo stato del ticket Jira in. `ACCEPTED RISK` Questo chiude il ticket Jira.
8. Un evento CloudWatch giornaliero avvia la funzione di aggiornamento Lambda, che identifica i ticket Jira chiusi e aggiorna i relativi risultati CSPM di Security Hub come. `SUPPRESSED`

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita in tutte le regioni. Account AWS
- Amazon CloudWatch Events ti aiuta a monitorare gli eventi di sistema per AWS le tue risorse utilizzando regole per abbinare gli eventi e indirizzarli verso funzioni o stream.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS Organizationsè un servizio di gestione degli account che ti aiuta a consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente.
- Con Gestione dei segreti AWS puoi sostituire le credenziali nel codice,incluse le password, con una chiamata API a Secrets Manager in modo da recuperare il segreto a livello di codice.
- AWS Security Hub CSPMfornisce una visione completa dello stato di sicurezza in AWS. Inoltre, consente di verificare la conformità AWS dell'ambiente agli standard e alle best practice del settore della sicurezza.

Archivio di codice

Il codice per questo pattern è disponibile su GitHub, nel repository aws-securityhub-jira-software-integration. Include il codice di esempio e il flusso di lavoro Jira per questa soluzione.

Epiche

Configura Jira

Operazione	Description	Competenze richieste
Importa il flusso di lavoro.	In qualità di amministratore di Jira, importa il `issue-workflow.xml` file nella tua istanza di Jira Data Center. Se usi Jira Cloud, devi creare il flusso di lavoro in base ai file and. `assets/jira-cloud-transitions.png` `assets/jira-cloud-status.png` I file possono essere trovati nel repository aws-securityhub-jira-software-integration in. GitHub Per istruzioni, consulta Usare XML per creare un flusso di lavoro (documentazione di Jira).	Amministratore Jira
Attiva e assegna il flusso di lavoro.	I flussi di lavoro sono inattivi finché non vengono assegnati a uno schema di workflow. Quindi si assegna lo schema di flusso di lavoro a un progetto. 1. Per il tuo progetto, assicurati di aver identificato uno schema di tipi di problema per il progetto. Puoi creare un nuovo tipo di problema o	Amministratore Jira

Operazione	Description	Competenze richieste
	selezionarne uno esistente, ad esempioBug. 2. Assegna il flusso di lavoro importato a uno schema di flusso di lavoro in base alle istruzioni in Attivazione di un flusso di lavoro (documentazione di Jira). 3. Assegna lo schema di flusso di lavoro a un progetto in base alle istruzioni in Associare uno schema di flusso di lavoro a un progetto (documentazione Jira).	

Imposta i parametri della soluzione

Operazione	Description	Competenze richieste
Configura i parametri della soluzione.	1. Nella cartella conf, apri`params_prod.shfile` . 2. Fornite i valori per i seguenti parametri: • `ORG_ACCOUNT_ID` — L'ID dell'account AWS Organizations di gestione. La soluzione legge i tag dell'account e assegna i ticket agli specifici contatti di	Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	sicurezza definiti in tali Account AWS tag. • `ORG_ROLE`— Il nome del ruolo IAM utilizzato per accedere all'account di AWS Organizations gestione. Questo ruolo deve disporre di `OrganizationsReadOnlyAccess` autorizzazioni. • `EXTERNAL_ID` — Un parametro opzionale se si utilizza un ID esterno per assumere il ruolo IAM definito in`ORG_ROLE`. Per ulteriori informazioni, consulta Come utilizzare un ID esterno (documentazione IAM). • `JIRA_DEFAULT_ASSIGNEE` — Questo è l'assegnatario predefinito di Jira per tutti i problemi di sicurezza. Questo assegnatario predefinito viene utilizzato nel caso in cui l'account non sia etichettato correttamente o non sia possibile assumere il ruolo. • `JIRA_INSTANCE` — L'indirizzo HTTPS del tuo endpoint Jira nel	

Operazione	Description	Competenze richieste
	seguente formato: `team-<team-id>.atlassian.net/` • `JIRA_PROJECT_KEY` — Il nome della chiave del progetto Jira utilizzata per creare i ticket, ad esempio o. `SEC` `TEST` Questo progetto deve già esistere in Jira. • `ISSUE_TYPE` — Il nome dello schema del tipo di problema assegnato al progetto in Jira, ad esempio `Bug` o. `Security Issue` • `REGIONS`— Elenco dei Regione AWS codici in cui si desidera implement are questa soluzione, ad esempio. `eu-west-1` 3. Salvate e chiudete il file dei parametri della soluzione.	

Operazione	Description	Competenze richieste
Identifica i risultati che desideri automatizzare.	1. Apri la console Security Hub CSPM. 2. Nel riquadro di navigazione CSPM di Security Hub, scegli Findings. 3. Scegli il titolo del risultato. 4. Scegli l'ID del ritrovamento. Viene visualizzato il codice JSON completo per il risultato. 5. Nel JSON, copia la stringa nel `GeneratorId` campo. Questo valore è in AWS Security Finding Format (ASFF). Ad esempio, `aws-foundational-security-best-practices/v/1.0.0/S3.1` corrisponde ai risultati del controllo di sicurezza S3.1, deve essere abilitata l'impostazione S3 Block Public Access. 6. Ripeti questi passaggi finché non hai copiato tutti i `GeneratorID` valori dei risultati che desideri automatizzare.	

Operazione	Description	Competenze richieste
Aggiungi i risultati al file di configurazione.	1. In src/code, apri il `config.jsonconfig` file. 2. Incolla i `GeneratorID` valori recuperati nella storia precedente nel `default` parametro e usa le virgole per separare ogni ID. 3. Salva e chiudi il file di configurazione . Il seguente esempio di codice mostra l'automazione dei risultati e. `aws-found ational-security-b est-practices/v/1. 0.0/SNS.1 aws-found ational-security-b est-practices/v/1. 0.0/S3.1` `{ "Controls" : { "eu-west-1": [ "arn:aws: securityhub:::rule set/cis-aws-founda tions-benchmark/v/ 1.2.0/rule/1.22" ], "default": [ aws-foundational- security-best-pra ctices/v/1.0.0/SNS.1,`	Amministratore di sistema AWS

Operazione	Description	Competenze richieste

	<pre>aws-foundational-security-best-practices/v/1.0.0/S3.1
	     ]
	    }
	 }</pre>
	Note
	Puoi scegliere di automatizzare risultati diversi per ciascuno di essi. Regione AWS Una buona pratica per evitare la duplicazione dei risultati consiste nel selezionare una singola regione per automatizzare la creazione di controlli relativi a IAM.	

Implementa l'integrazione

Operazione	Description	Competenze richieste
Implementa l'integrazione.	In un terminale a riga di comando, inserisci il seguente comando: <pre>./deploy.sh prod</pre>	Amministratore di sistema AWS
Carica le credenziali di Jira su Secrets Manager.	1. Apri la console Secrets Manager.	Amministratore di sistema AWS

Operazione	Description	Competenze richieste
	2. In Secrets (Segreti), scegli Store a new secret (Archivia un nuovo segreto). 3. Per Secret type (Tipo di segreto), scegli Other type of secret (Altro tipo di segreto). 4. Se utilizzi Jira Enterprise, per le coppie chiave/valore, procedi come segue: • Nella prima riga, inserisci `auth` nella casella chiave, quindi inserisci `token_auth` nella casella del valore. • Aggiungi una seconda riga, inserisci `token` nella casella chiave, quindi inserisci il tuo token di accesso personale nella casella del valore. Se utilizzi Jira Cloud, per le coppie chiave/valore, procedi come segue: • Nella prima riga, inserisci `auth` nella casella chiave, quindi inserisci `basic_auth` nella casella del valore. • Aggiungi una seconda riga, inseriscila `token` nella casella chiave, quindi inserisci il tuo	

Operazione	Description	Competenze richieste
	token API nella casella del valore. • Aggiungi una terza riga, inserisci `email` nella casella chiave, quindi inserisci il tuo indirizzo e-mail nella casella del valore. 5. Scegli Next (Successivo). 6. Per Nome segreto, immetteteJira-Token , quindi nella parte inferiore della pagina, scegliete Avanti. 7. Nella pagina Secret rotation (Rotazione del segreto), mantieni Disable automatic rotation (Disabilita la rotazione automatica) e quindi nella parte inferiore della pagina, scegli Next (Successivo). 8. Nella pagina Review (Revisione), rivedi i dettagli dei segreti e quindi scegli Store (Archivia).	

Operazione	Description	Competenze richieste
Crea l'azione personalizzata Security Hub CSPM.	1. Per ciascuna di esse Regione AWS, in AWS Command Line Interface (AWS CLI), utilizzat e il create-action-targetcomando per creare un'azione personalizzata Security Hub CSPM denominata. `CreateJiraIssue` `aws securityhub create-action-target --name "CreateJiraIssue" \ --description "Create ticket in JIRA" \ --id "CreateJiraIssue" --region $<aws-region>` 2. Apri la console Security Hub CSPM. 3. Nel riquadro di navigazione CSPM di Security Hub, scegli Findings. 4. Nell'elenco dei risultati, seleziona i risultati che desideri aumentare. 5. Nel menu Azioni, scegli. `CreateJiraIssue`	Amministratore di sistema AWS

Risorse correlate

- AWS Connettore di gestione dei servizi per Jira Service Management
- AWS Standard di best practice di sicurezza di base

Informazioni aggiuntive

Esempio di ticket Jira

Quando si verifica una ricerca CSPM di Security Hub specificata, questa soluzione crea automaticamente un ticket Jira. Il ticket include le seguenti informazioni:

- Titolo: il titolo identifica il problema di sicurezza nel seguente formato:

```
AWS Security Issue :: <AWS account ID> :: <Security Hub finding title>
```

- Descrizione: la sezione descrittiva del ticket descrive il controllo di sicurezza associato al risultato, include un collegamento al risultato nella console CSPM di Security Hub e fornisce una breve descrizione di come gestire il problema di sicurezza nel flusso di lavoro di Jira.

Di seguito è riportato un esempio di ticket Jira generato automaticamente.

Titolo	AWS Problema di sicurezza:: 012345678912:: Le politiche delle funzioni Lambda dovrebbero vietare l'accesso pubblico.
Descrizione	Qual è il problema? Abbiamo rilevato un problema di sicurezza all'interno dello Account AWS 012345678912 di cui sei responsabile. Questo controllo verifica se la politica delle AWS Lambda funzioni allegata alla risorsa Lambda proibisce l'accesso pubblico. Se la policy della funzione Lambda consente l'accesso pubblico, il controllo fallisce. <Link to Security Hub CSPM finding>

Cosa devo fare con il biglietto?

- Accedi all'account e verifica la configura zione. Conferma di aver lavorato sul ticket spostandolo su «Allocated for Fix». Una volta risolto, passa alla versione di prova, in modo che la sicurezza confermi che il problema venga risolto.
- Se ritieni che il rischio debba essere accettato, spostalo in «In attesa di accettazi one del rischio». Ciò richiederà la revisione da parte di un tecnico della sicurezza.
- Se pensi che sia un falso positivo, sostituis cilo a «Contrassegna come falso positivo». Questo verrà esaminato da un tecnico della sicurezza e reopened/closed di conseguen za.

Crea una pipeline per immagini di container rinforzate utilizzando Image EC2 Builder e Terraform

Mike Saintcross e Andrew Ranes, Amazon Web Services

Riepilogo

Questo modello crea una pipeline di Image Builder che produce un'EC2 immagine del contenitore di base Amazon Linux 2 rinforzata. Terraform viene utilizzato come strumento Infrastructure as Code (IaC) per configurare e fornire l'infrastruttura utilizzata per creare immagini di container rinforzate. La ricetta ti aiuta a distribuire un'immagine di container Amazon Linux 2 basata su Docker che è stata rafforzata secondo Red Hat Enterprise Linux (RHEL) 7 STIG Version 3 Release 7 – Medium. (Vedere la STIG-Build-Linux-Medium versione 2022.2.1 nella sezione dei componenti Linux STIG della documentazione di EC2 Image Builder.) Questa viene definita immagine dorata del contenitore.

La build include due EventBridge regole Amazon. Una regola avvia la pipeline di immagini del contenitore quando il risultato di Amazon Inspector è alto o critico, in modo che le immagini non sicure vengano sostituite. Questa regola richiede l'abilitazione della scansione avanzata di Amazon Inspector e Amazon Elastic Container Registry (Amazon ECR). L'altra regola invia notifiche a una coda Amazon Simple Queue Service (Amazon SQS) dopo che un'immagine è stata inviata con successo al repository Amazon ECR, per aiutarti a utilizzare le immagini più recenti dei container.

> **Note**
>
> Amazon Linux 2 sta per terminare il supporto. Per ulteriori informazioni, consulta Amazon Linux 2 FAQs.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS in cui puoi implementare l'infrastruttura.
- AWS Command Line Interface (AWS CLI) installata per impostare le credenziali AWS per la distribuzione locale.
- Terraform è stato scaricato e configurato seguendo le istruzioni nella documentazione di Terraform.
- Git (se stai effettuando il provisioning da una macchina locale).

- Un ruolo all'interno dell'account AWS che puoi utilizzare per creare risorse AWS.
- Tutte le variabili definite nel file.tfvars. Oppure puoi definire tutte le variabili quando applichi la configurazione Terraform.

Limitazioni

- Questa soluzione crea un'infrastruttura Amazon Virtual Private Cloud (Amazon VPC) che include un gateway NAT e un gateway Internet per la connettività Internet dalla sua sottorete privata. Non puoi utilizzare gli endpoint VPC, perché il processo di bootstrap di AWS Task Orchestrator ed Executor () AWSTOE installa la versione 2 dell'interfaccia a riga di comando di AWS da Internet.

Versioni del prodotto

- Amazon Linux 2
- AWS CLI versione 1.1 o successiva

Architecture

Stack tecnologico Target

Questo modello crea 43 risorse, tra cui:

- Due bucket Amazon Simple Storage Service (Amazon S3): uno per i file dei componenti della pipeline e uno per l'accesso al server e i log di flusso di Amazon VPC
- Un repository Amazon ECR
- Un cloud privato virtuale (VPC) che contiene una sottorete pubblica, una sottorete privata, tabelle di routing, un gateway NAT e un gateway Internet
- Una pipeline, una ricetta e componenti di EC2 Image Builder
- Un'immagine del contenitore
- Una chiave AWS Key Management Service (AWS KMS) per la crittografia delle immagini
- Una coda SQS
- Tre ruoli: uno per eseguire la pipeline di EC2 Image Builder, un profilo di istanza per EC2 Image Builder e uno per le regole EventBridge
- Due regole EventBridge

Struttura del modulo Terraform

Per il codice sorgente, vedere il GitHub repository Terraform Image EC2 Builder Container Hardening Pipeline.

```
### components.tf
### config.tf
### dist-config.tf
### files
#   ###assumption-policy.json
### hardening-pipeline.tfvars
### image.tf
### infr-config.tf
### infra-network-config.tf
### kms-key.tf
### main.tf
### outputs.tf
### pipeline.tf
### recipes.tf
### roles.tf
### sec-groups.tf
### trigger-build.tf
### variables.tf
```

Dettagli del modulo

- `components.tf`contiene una risorsa di caricamento Amazon S3 per caricare il contenuto della `/files` directory. Qui puoi anche aggiungere in modo modulare file YAML di componenti personalizzati.
- `/files`contiene i `.yml` file che definiscono i componenti utilizzati in. `components.tf`
- `image.tf`contiene le definizioni per il sistema operativo con immagine di base. Qui è possibile modificare le definizioni per una diversa pipeline di immagini di base.
- `infr-config.tf`e `dist-config.tf` contengono le risorse per l'infrastruttura AWS minima necessaria per avviare e distribuire l'immagine.
- `infra-network-config.tf`contiene l'infrastruttura VPC minima in cui distribuire l'immagine del contenitore.
- `hardening-pipeline.tfvars`contiene le variabili Terraform da utilizzare al momento dell'applicazione.
- `pipeline.tf`crea e gestisce una pipeline EC2 Image Builder in Terraform.

- `recipes.tf`è dove puoi specificare diverse miscele di componenti per creare ricette di contenitori.
- `roles.tf`contiene le definizioni delle policy di AWS Identity and Access Management (IAM) per il profilo dell'istanza Amazon Elastic Compute Cloud (Amazon EC2) e il ruolo di distribuzione della pipeline.
- `trigger-build.tf`contiene EventBridge le regole e le risorse di coda SQS.

Architettura Target

Il diagramma illustra il seguente flusso di lavoro:

1. EC2 Image Builder crea un'immagine del contenitore utilizzando la ricetta definita, che installa gli aggiornamenti del sistema operativo e applica RHEL Medium STIG all'immagine di base di Amazon Linux 2.
2. L'immagine protetta viene pubblicata in un registro Amazon ECR privato e una EventBridge regola invia un messaggio a una coda SQS quando l'immagine è stata pubblicata correttamente.
3. Se Amazon Inspector è configurato per una scansione avanzata, esegue la scansione del registro Amazon ECR.
4. Se Amazon Inspector genera un risultato di gravità critica o elevata per l'immagine, una EventBridge regola attiva la pipeline di Image Builder per riavviare la pipeline di EC2 Image Builder e pubblicare un'immagine appena protetta.

Automazione e scalabilità

- Questo modello descrive come effettuare il provisioning dell'infrastruttura e creare la pipeline sul computer. Tuttavia, è destinato a essere utilizzato su larga scala. Invece di distribuire i moduli Terraform localmente, puoi utilizzarli in un ambiente multi-account, come un ambiente AWS Control Tower con Account Factory for Terraform. In tal caso, è necessario utilizzare un bucket S3 con stato di backend per gestire i file di stato Terraform anziché gestire lo stato di configurazione localmente.
- Per un utilizzo scalabile, distribuisci la soluzione su un account centrale, ad esempio un account Shared Services o Common Services, da un modello di account Control Tower o landing zone e concedi agli account dei consumatori l'autorizzazione ad accedere al repository Amazon ECR e alla chiave AWS KMS. Per ulteriori informazioni sulla configurazione, consulta l'articolo Re:post Come posso consentire a un account secondario di inviare o estrarre immagini nel mio archivio di immagini Amazon ECR? Ad esempio, in un distributore automatico di account o Account

Factory for Terraform, aggiungi le autorizzazioni a ogni linea di base dell'account o alla baseline di personalizzazione dell'account per fornire l'accesso al repository Amazon ECR e alla chiave di crittografia.

- Dopo aver distribuito la pipeline di immagini del contenitore, puoi modificarla utilizzando le funzionalità di Image EC2 Builder come i componenti, che ti aiutano a impacchettare più componenti nella build Docker.
- La chiave AWS KMS utilizzata per crittografare l'immagine del contenitore deve essere condivisa tra gli account in cui è destinata l'immagine.
- Puoi aggiungere il supporto per altre immagini duplicando l'intero modulo Terraform e modificando i seguenti attributi: `recipes.tf`
  - Modifica su un altro `parent_image = "amazonlinux:latest"` tipo di immagine.
  - Modifica `repository_name` in modo che punti a un repository Amazon ECR esistente. Questo crea un'altra pipeline che distribuisce un tipo di immagine principale diverso nel tuo repository Amazon ECR esistente.

Tools (Strumenti)

Strumenti

- Terraform (fornitura IaC)
- Git (se si effettua il provisioning locale)
- AWS CLI versione 1 o versione 2 (se il provisioning è locale)

Codice

Il codice per questo pattern si trova nel GitHub repository Terraform Image EC2 Builder Container Hardening Pipeline. Per utilizzare il codice di esempio, segui le istruzioni nella sezione successiva.

Epiche

Fornisci l'infrastruttura

Operazione	Description	Competenze richieste
Imposta le credenziali locali.	Configura le tue credenziali temporanee AWS.	AWS DevOps

Operazione	Description	Competenze richieste

	1. Verifica se la CLI AWS è installata:
	
	`$ aws --version`
	`aws-cli/1.16.249`
	`Python/3.6.8...`
	
	• La versione AWS CLI deve essere 1.1 o successiva.
	• Se il comando non viene trovato, installa l'AWS CLI.
	2. Esegui `aws configure` e fornisci i seguenti valori:
	
	`$ aws configure`
	`AWS Access Key ID [**********xxxx]: <Your AWS access key ID>`
	`AWS Secret Access Key [**********xxxx]: <Your AWS secret access key>`
	`Default region name: [us-east-1]: <Your desired Region for deployment>`
	`Default output format [None]: <Your desired output format>`	

Operazione	Description	Competenze richieste

Clonare il repository.	1. Clona il repository fornito con questo modello. Puoi usare HTTPS o Secure Shell (SSH).
	
	HTTPS:
	
	`git clone https://github.com/aws-samples/terraform-ec2-image-builder-container-hardening-pipeline`
	
	SSH:
	
	`git clone git@github.com:aws-samples/terraform-ec2-image-builder-container-hardening-pipeline.git`
	2. Passa alla directory locale che contiene questa soluzione:
	
	`cd terraform-ec2-image-builder-container-hardening-pipeline`
		AWS DevOps

Operazione	Description	Competenze richieste

Aggiorna le variabili.	Aggiorna le variabili nel `hardening-pipeline.tfvars` file in modo che corrispondano all'ambiente e alla configurazione desiderata. È necessario fornire il proprio `account_id`. Tuttavia, è necessario modificare anche il resto delle variabili per adattarle alla distribuzione desiderata. Tutte le variabili sono obbligatorie.
	
	`account_id = "<DEPLOYMENT-ACCOUNT-ID>"`
	`aws_region = "us-east-1"`
	`vpc_name = "example-hardening-pipeline-vpc"`
	`kms_key_alias = "image-builder-container-key"`
	`ec2_iam_role_name = "example-hardening-instance-role"`
	`hardening_pipeline_role_name = "example-hardening-pipeline-role"`
	`aws_s3_ami_resources_bucket = "example-hardening-ami-resources-bucket-0123"`
	`image_name = "example-hardening-al2-container-image"`
		AWS DevOps

Operazione	Description	Competenze richieste
	ecr_name = "example-hardening-container-repo" recipe_version = "1.0.0" ebs_root_vol_size = 10 Ecco una descrizione di ogni variabile: • account_id – Il numero di account AWS in cui desideri distribuire la soluzione. • aws_region – La regione AWS in cui desideri implementare la soluzione. • vpc_name– Il nome dell'infrastruttura VPC. • kms_key_alias – Il nome della chiave AWS KMS da utilizzare per la configurazione dell'infrastruttura Image EC2 Builder. • ec2_iam_role_name – Il nome del ruolo che verrà utilizzato come profilo dell' EC2 istanza. • hardening_pipeline_role_name – Il nome del ruolo che verrà utilizzato per implementare la pipeline di rafforzamento. • aws_s3_ami_resources_bucket – Il nome di un bucket S3 che ospiterà	

Operazione	Description	Competenze richieste
	tutti i file necessari per creare le immagini della pipeline e del contenitore. • `image_name` – Il nome dell'immagine del contenitore. Questo valore deve essere compreso tra 3 e 50 caratteri e deve contenere solo caratteri alfanumerici e trattini. • `ecr_name`– Il nome del registro Amazon ECR in cui archiviare le immagini del contenitore. • `recipe_version` – La versione della ricetta dell'immagine. Il valore predefinito è 1.0.0. • `ebs_root_vol_size` – La dimensione (in gigabyte) del volume root di Amazon Elastic Block Store (Amazon EBS). Il valore predefinito è 10 gigabyte.	

Operazione	Description	Competenze richieste
Inizializza Terraform.	Dopo aver aggiornato i valori delle variabili, puoi inizializzare la directory di configurazione Terraform. L'inizializzazione di una directory di configurazione scarica e installa il provider AWS, definito nella configura zione. `terraform init` Dovresti vedere un messaggio che dice che Terraform è stato inizializzato con successo e identifica la versione del provider che è stata installata.	AWS DevOps
Implementa l'infrastruttura e crea un'immagine del contenitore.	Usa il seguente comando per inizializzare, convalidare e applicare i moduli Terraform all'ambiente utilizzando le variabili definite nel file: `.tfvars` `terraform init && terraform validate && terraform apply -var-file *.tfvars -auto-approve`	AWS DevOps

Operazione	Description	Competenze richieste
Personalizza il contenitore.	È possibile creare una nuova versione di una ricetta contenitore dopo che EC2 Image Builder ha distribuito la pipeline e la ricetta iniziale. È possibile aggiungere uno qualsiasi degli oltre 31 componenti disponibili in EC2 Image Builder per personalizzare la build del contenitore. Per ulteriori informazioni, vedere la sezione Componenti di Creare una nuova versione di una ricetta contenitore nella documentazione di EC2 Image Builder.	Amministratore AWS

Convalida le risorse

Operazione	Description	Competenze richieste
Convalida il provisioning dell'infrastruttura AWS.	Dopo aver completato con successo il primo `apply` comando Terraform, se stai effettuando il provisioning localmente, dovresti vedere questo frammento nel terminale del tuo computer locale: `Apply complete! Resources: 43 added, 0 changed, 0 destroyed.`	AWS DevOps

Operazione	Description	Competenze richieste
Convalida le singole risorse dell'infrastruttura AWS.	Per convalidare le singole risorse che sono state distribuite, se esegui il provisioning a livello locale, puoi eseguire il seguente comando: `terraform state list` Questo comando restituisce un elenco di 43 risorse.	AWS DevOps

Rimuovi risorse

Operazione	Description	Competenze richieste
Rimuovi l'infrastruttura e l'immagine del contenitore.	Quando hai finito di lavorare con la configurazione di Terraform, puoi eseguire il seguente comando per rimuovere le risorse: `terraform init && terraform validate && terraform destroy -var-file *.tfvars -auto-approve`	AWS DevOps

risoluzione dei problemi

Problema	Soluzione
Errore durante la convalida delle credenziali del provider	Quando esegui Terraform `apply` o il `destroy` comando dal tuo computer locale, potresti riscontrare un errore simile al seguente: `Error: configuring Terraform AWS Provider: error validating provider credentials: error calling sts:GetCallerIdentity: operation error STS: GetCallerIdentity, https response error StatusCode: 403, RequestID: 123456a9-fbc1-40ed-b8d8-513d0133ba7f, api error InvalidClientTokenId: The security token included in the request is invalid.` Questo errore è causato dalla scadenza del token di sicurezza per le credenziali utilizzate nella configurazione del computer locale. Per risolvere l'errore, consulta Impostare e visualizzare le impostazioni di configurazione nella documentazione dell'interfaccia a riga di comando di AWS.

Risorse correlate

- Terraform EC2 Image Builder Container Hardening Pipeline (repository) GitHub
- EC2 Documentazione di Image Builder
- AWS Control Tower Account Factory per Terraform (post sul blog AWS)
- Bucket S3 con stato di backend (documentazione Terraform)
- Installazione o aggiornamento della versione più recente dell'interfaccia a riga di comando di AWS (documentazione dell'interfaccia a riga di comando di AWS)

- Scarica Terraform

Centralizza la gestione delle chiavi di accesso IAM in AWS Organizations utilizzando Terraform

Aarti Rajput, Chintamani Aphale, TV.R.L.Phani Kumar Dadi, Pratap Kumar Nanda, Pradip kumar Pandey e Mayuri Shinde, Amazon Web Services

Riepilogo

L'applicazione delle regole di sicurezza per chiavi e password è un compito essenziale per ogni organizzazione. Una regola importante è ruotare le chiavi AWS Identity and Access Management (IAM) a intervalli regolari per rafforzare la sicurezza. Le chiavi di accesso AWS vengono generalmente create e configurate localmente ogni volta che i team desiderano accedere ad AWS dall'interfaccia a riga di comando di AWS (AWS CLI) o da applicazioni esterne ad AWS. Per mantenere una forte sicurezza in tutta l'organizzazione, le vecchie chiavi di sicurezza devono essere modificate o eliminate dopo che il requisito è stato soddisfatto o a intervalli regolari. Il processo di gestione delle rotazioni delle chiavi tra più account in un'organizzazione è lungo e noioso. Questo modello consente di automatizzare il processo di rotazione utilizzando i servizi Account Factory for Terraform (AFT) e AWS.

Il modello offre i seguenti vantaggi:

- Gestisce la chiave di accesso IDs e le chiavi di accesso segrete su tutti gli account dell'organizzazione da una posizione centrale.
- Ruota automaticamente le variabili `AWS_ACCESS_KEY_ID` e di `AWS_SECRET_ACCESS_KEY` ambiente.
- Impone il rinnovo se le credenziali dell'utente sono compromesse.

Il modello utilizza Terraform per distribuire funzioni AWS Lambda, regole EventBridge Amazon e ruoli IAM. Una EventBridge regola viene eseguita a intervalli regolari e chiama una funzione Lambda che elenca tutte le chiavi di accesso utente in base a quando sono state create. Le funzioni Lambda aggiuntive creano un nuovo ID chiave di accesso e una chiave di accesso segreta, se la chiave precedente è più vecchia del periodo di rotazione definito (ad esempio, 45 giorni), e avvisano un amministratore della sicurezza utilizzando Amazon Simple Notification Service (Amazon SNS) e Amazon Simple Email Service (Amazon SES). I segreti vengono creati in AWS Secrets Manager per quell'utente, la vecchia chiave di accesso segreta viene archiviata in Secrets Manager e le autorizzazioni per l'accesso alla vecchia chiave sono configurate. Per garantire che la vecchia chiave di accesso non venga più utilizzata, viene disabilitata dopo un periodo di inattività (ad esempio,

60 giorni, ovvero 15 giorni dopo la rotazione delle chiavi nel nostro esempio). Dopo un periodo di buffer inattivo (ad esempio, 90 giorni o 45 giorni dopo la rotazione delle chiavi nel nostro esempio), le vecchie chiavi di accesso vengono eliminate da AWS Secrets Manager. Per un'architettura e un flusso di lavoro dettagliati, consulta la sezione Architettura.

Prerequisiti e limitazioni

- Una landing zone per la tua organizzazione creata utilizzando AWS Control Tower (versione 3.1 o successiva)
- Account Factory for Terraform (AFT) configurato con tre account:
  - L'account di gestione dell'organizzazione gestisce l'intera organizzazione da una posizione centrale.
  - L'account di gestione AFT ospita la pipeline Terraform e distribuisce l'infrastruttura nell'account di distribuzione.
  - L'account di implementazione implementa questa soluzione completa e gestisce le chiavi IAM da una posizione centrale.
- Terraform versione 0.15.0 o successiva per il provisioning dell'infrastruttura nell'account di distribuzione.
- Un indirizzo e-mail configurato in Amazon Simple Email Service (Amazon SES).
- (Consigliato) Per migliorare la sicurezza, implementa questa soluzione all'interno di una sottorete privata (account di distribuzione) all'interno di un cloud privato virtuale (VPC). Puoi fornire i dettagli del VPC e della sottorete quando personalizzi le variabili (vedi Personalizzare i parametri per la pipeline del codice nella sezione Epics).

Architecture

Archivi AFT

Questo modello utilizza Account Factory for Terraform (AFT) per creare tutte le risorse AWS richieste e la pipeline di codice per distribuire le risorse in un account di distribuzione. La pipeline di codice viene eseguita in due repository:

- La personalizzazione globale contiene il codice Terraform che verrà eseguito su tutti gli account registrati con AFT.
- Le personalizzazioni dell'account contengono il codice Terraform che verrà eseguito nell'account di distribuzione.

Dettagli delle risorse

I CodePipeline lavori AWS creano le seguenti risorse nell'account di distribuzione:

- EventBridge Regola AWS e regola configurata
- Funzione Lambda `account-inventory`
- Funzione Lambda `IAM-access-key-rotation`
- Funzione Lambda `Notification`
- Bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) che contiene un modello di e-mail
- Policy IAM richiesta

Architettura

Il diagramma illustra quanto segue:

1. Una EventBridge regola chiama la funzione `account-inventory` Lambda ogni 24 ore.
2. La funzione `account-inventory` Lambda richiede ad AWS Organizations un elenco di tutti gli account AWS IDs, i nomi degli account e le e-mail degli account.
3. La funzione `account-inventory` Lambda avvia una funzione `IAM-access-key-auto-rotation` Lambda per ogni account AWS e gli trasmette i metadati per un'ulteriore elaborazione.
4. La funzione `IAM-access-key-auto-rotation` Lambda utilizza un ruolo IAM presunto per accedere all'account AWS. Lo script Lambda esegue un controllo su tutti gli utenti e sulle relative chiavi di accesso IAM nell'account.
5. La soglia di rotazione delle chiavi IAM (periodo di rotazione) viene configurata come variabile di ambiente quando viene implementata la funzione `IAM-access-key-auto-rotation` Lambda. Se il periodo di rotazione viene modificato, la funzione `IAM-access-key-auto-rotation` Lambda viene ridistribuita con una variabile di ambiente aggiornata. Puoi configurare i parametri per impostare il periodo di rotazione, il periodo di inattività per le vecchie chiavi e il buffer inattivo, dopodiché le vecchie chiavi verranno eliminate (vedi Personalizzare i parametri per la pipeline del codice nella sezione Epics).
6. La funzione `IAM-access-key-auto-rotation` Lambda convalida l'età della chiave di accesso in base alla sua configurazione. Se l'età della chiave di accesso IAM non ha superato il periodo di rotazione definito, la funzione Lambda non esegue ulteriori azioni.

7. Se l'età della chiave di accesso IAM ha superato il periodo di rotazione definito, la funzione `IAM-access-key-auto-rotation` Lambda crea una nuova chiave e ruota la chiave esistente.
8. La funzione Lambda salva la vecchia chiave in Secrets Manager e limita le autorizzazioni all'utente le cui chiavi di accesso si discostano dagli standard di sicurezza. La funzione Lambda crea anche una policy basata sulle risorse che consente solo al principale IAM specificato di accedere e recuperare il segreto.
9. La funzione `IAM-access-key-rotation` Lambda chiama la funzione Lambda`Notification`.
10. La funzione `Notification` Lambda interroga il bucket S3 per un modello di e-mail e genera dinamicamente messaggi e-mail con i metadati delle attività pertinenti.
11. La funzione `Notification` Lambda richiama Amazon SES per ulteriori azioni.
12. Amazon SES invia un'e-mail all'indirizzo e-mail del proprietario dell'account con le informazioni pertinenti.

Tools (Strumenti)

Servizi AWS

- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle. Questo modello richiede ruoli e autorizzazioni IAM.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.
- AWS Secrets Manager ti aiuta a sostituire le credenziali codificate nel codice, comprese le password, con una chiamata API a Secrets Manager per recuperare il segreto a livello di codice.
- Amazon Simple Email Service (Amazon SES) Simple Email Service (Amazon SES) ti aiuta a inviare e ricevere e-mail utilizzando i tuoi indirizzi e-mail e domini.

Altri strumenti

- Terraform è uno strumento di infrastruttura come codice (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.

Archivio di codici

Le istruzioni e il codice per questo modello sono disponibili nell'archivio di rotazione delle chiavi di accesso GitHub IAM. Puoi distribuire il codice nell'account di distribuzione centrale AWS Control Tower per gestire la rotazione delle chiavi da una posizione centrale.

Best practice

- Per IAM, consulta le best practice di sicurezza nella documentazione IAM.
- Per la rotazione delle chiavi, consulta le linee guida per l'aggiornamento delle chiavi di accesso nella documentazione IAM.

Poemi epici

Configura i file sorgente

Operazione	Description	Competenze richieste

Clonare il repository.	1. Clona il GitHub repository di rotazione delle chiavi di accesso IAM:
	`$ git clone https://github.com/aws-samples/centralized-iam-key-management-aws-organizations-terraform.git`
	2. Verifica che la copia locale del repository contenga tre cartelle:
	`$ cd Iam-Access-keys-Rotation`
	`$ ls`
	`org-account-customization`
	`global-account-customization`
		DevOps ingegnere

Operazione	Description	Competenze richieste
	`account-custom ization`	

Configurazione degli account

Operazione	Description	Competenze richieste

Configura l'account di bootstrap.	Come parte del processo di bootstrap AFT, dovresti avere una cartella chiamata `aft-bootstrap` sul tuo computer locale.
	1. Copia manualmente tutti i file Terraform dalla cartella locale alla GitHub org-account-customizationcartella. `aft-bootstrap`
	2. Esegui i comandi Terraform per configurare il ruolo globale tra account nell'account di gestione AWS Control Tower:
	`$ cd aft-bootstrap`
	`$ terraform init`
	`$ terraform apply –auto-approve`
		DevOps ingegnere

Configura personalizzazioni globali.	Come parte della configurazione della cartella AFT, è necessario che una cartella venga richiamata `aft-globa`	DevOps ingegnere

Operazione	Description	Competenze richieste

	`l-customizations` sul computer locale.
	
	1. Copia manualmente tutti i file Terraform dalla GitHub global-account-customizationcartella locale alla `aft-global-customizations/terraform` cartella.
	2. Invia il codice ad AWS CodeCommit:
	
	`$ git add *`
	`$ git commit -m "message"`
	`$ git push`	

Operazione	Description	Competenze richieste

Configura le personalizzazioni dell'account.	Come parte della configurazione della cartella AFT, è necessaria una cartella chiamata `aft-account-customizations` sul computer locale.
	1. Crea una cartella con il tuo numero di account fornito.
	2. Copia manualmente tutti i file Terraform dalla cartella locale di GitHub personalizzazione dell'account alla tua cartella. `aft-account-customizations/<vended account>/terraform`
	3. Invia il codice ad AWS CodeCommit:
	`$ git add *`
	`$ git commit -m "message"`
	`$ git push`
		DevOps ingegnere

Personalizza i parametri per la pipeline del codice

Operazione	Description	Competenze richieste
Personalizza i parametri della pipeline di codice non Terraform per tutti gli account.	Crea un file chiamato `input.auto.tfvars` nella `aft-global-customizations/terraform/` cartella e fornisci i dati di input	DevOps ingegnere

Operazione	Description	Competenze richieste
	richiesti. Vedi il file nel GitHub repository per i valori predefiniti.	

Operazione	Description	Competenze richieste
Personalizza i parametri della pipeline di codice per l'account di distribuzione.	Crea un file chiamato `input.auto.tfvars` nella `aft-account-customizations/<AccountName>/terraform/` cartella e invia il codice ad AWS CodeCommit. L'invio di codice in AWS avvia CodeCommit automaticamente la pipeline di codice. Specificate i valori per i parametri in base ai requisiti della vostra organizzazione, tra cui i seguenti (consultate il file nel repository Github per i valori predefiniti): • `s3_bucket_name` — Un nome bucket univoco per il modello di email. • `s3_bucket_prefix` — Un nome di cartella all'interno del bucket S3. • `admin_email_address` — L'indirizzo e-mail dell'amministratore che dovrebbe ricevere la notifica. • `org_list_account` — Il numero di account dell'account di gestione. • `rotation_period` — Il numero di giorni dopo i quali	DevOps ingegnere

Operazione	Description	Competenze richieste
	una chiave deve essere ruotata da attiva a inattiva. • `inactive_period` — Il numero di giorni dopo i quali i tasti ruotati devono essere disattivati. Questo valore deve essere maggiore del valore di. `rotation_period` • `inactive_buffer` — Il periodo di tolleranza tra la rotazione e la disattivazione di una chiave. • `recovery_grace_period` — Il periodo di grazia tra la disattivazione e l'eliminazione di una chiave. • `dry_run_flag` — Imposta su true se desideri inviare una notifica all'amministratore a scopo di test, senza ruotare i tasti. • `store_secrets_in_central_account` — Imposta su true se desideri memorizzare il segreto nell'account di distribuzione. Se la variabile è impostata su false (impostazione predefinita), il segreto verrà archiviato nell'account del membro.	

Operazione	Description	Competenze richieste
	• `credential_replication_region` — La regione AWS in cui desideri distribuire la funzione Lambda e i bucket S3 per il modello di email. • `run_lambda_in_vpc` — Impostato su true per eseguire la funzione Lambda all'interno del VPC. • `vpc_id`— L'ID VPC dell'account di distribuzione, se si desidera eseguire la funzione Lambda all'interno del VPC. • `vpc_cidr`— L'intervallo CIDR per l'account di distribuzione. • `subnet_id` — La sottorete IDs per l'account di distribuzione. • `create_smtp_endpoint` — Imposta su true se desideri abilitare l'endpoint di posta elettronica.	

Convalida la rotazione dei tasti

Operazione	Description	Competenze richieste
Convalida la soluzione.	1. Dalla Console di gestione AWS, accedi all'account di distribuzione.	DevOps ingegnere

Operazione	Description	Competenze richieste
	2. Apri la console IAM e controlla se le credenziali utente (chiave di accesso IDs e chiavi segrete) vengono ruotate come specificato. 3. Dopo aver ruotato una chiave IAM, conferma quanto segue: • Il vecchio valore è archiviato in AWS Secrets Manager. • Il nome segreto è nel formato`Account_<account ID>_User_<username>_AccessKey` . • L'utente specificato nel `admin_email_address` parametro riceve una notifica via e-mail sulla rotazione dei tasti.	

Estendi la soluzione

Operazione	Description	Competenze richieste
Personalizza la data di notifica via e-mail.	Se desideri inviare notifiche e-mail in un giorno specifico prima di disabilitare la chiave di accesso, puoi aggiornare la funzione `IAM-access-key-`	DevOps ingegnere

Operazione	Description	Competenze richieste
	`auto-rotation` Lambda con le seguenti modifiche: 1. Definire una variabile chiamata`notify-period` . 2. Aggiungi una `if` condizione `main.py` prima di disattivare la chiave: `If (keyage>rotation-period-notify-period){ send_to_notifier(context, aws_account_id, account_name, resource_owner, resource_actions[resource_owner], dryrun, config.emailTemplateAudit) }`	

risoluzione dei problemi

Problema	Soluzione
Il job `account-inventory` Lambda non riesce `AccessDenied` durante l'elencazione degli account.	Se riscontri questo problema, devi convalidare le autorizzazioni: 1. Accedi all'account appena venduto, apri la CloudWatch console Amazon e quindi visualizza il gruppo `/aws/lambda/account-inventory-lambda` di CloudWatch log.

Problema	Soluzione
	2. Nei CloudWatch log più recenti, identifica il numero di account che causa il problema di accesso negato. 3. Accedi all'account di gestione AWS Control Tower e conferma che il ruolo `allow-list-account` è stato creato. 4. Se il ruolo non esiste, esegui nuovamente il codice Terraform utilizzando il comando. `terraform apply` 5. Scegli la scheda Account affidabile e verifica che lo stesso account sia attendibile.

Risorse correlate

- Pratiche consigliate da Terraform (documentazione Terraform)
- Le migliori pratiche di sicurezza in IAM (documentazione IAM)
- Le migliori pratiche per la rotazione delle chiavi (documentazione IAM)

Controlla una CloudFront distribuzione Amazon per la registrazione degli accessi, la versione HTTPS e TLS

SaiJeevan Devireddy e Bijesh Bal, Amazon Web Services

Riepilogo

Questo modello verifica una CloudFront distribuzione Amazon per assicurarsi che utilizzi HTTPS, utilizzi Transport Layer Security (TLS) versione 1.2 o successiva e che abbia la registrazione degli accessi abilitata. CloudFront è un servizio fornito da Amazon Web Services (AWS) che accelera la distribuzione di contenuti Web statici e dinamici, come .html, .css, .js e file di immagine, agli utenti. CloudFront fornisce i tuoi contenuti attraverso una rete mondiale di data center denominati edge location. Quando un utente richiede i contenuti che utilizzi CloudFront, la richiesta viene indirizzata all'edge location che offre la latenza (ritardo) più bassa, in modo che i contenuti vengano forniti con le migliori prestazioni possibili.

Questo modello fornisce una funzione AWS Lambda che viene avviata quando Amazon CloudWatch Events rileva la chiamata CloudFront CreateDistributionAPI, oppure. CreateDistributionWithTagsUpdateDistribution La logica personalizzata nella funzione Lambda valuta tutte le CloudFront distribuzioni create o aggiornate nell'account AWS. Invia una notifica di violazione utilizzando Amazon Simple Notification Service (Amazon SNS) se rileva le seguenti violazioni:

- Controlli globali:
  - Il certificato personalizzato non utilizza la versione TLS 1.2
  - La registrazione è disabilitata per la distribuzione
- Controlli di origine:
  - Origin non è configurato con la versione TLS 1.2
  - La comunicazione con l'origine è consentita su un protocollo diverso da HTTPS
- Controlli comportamentali:
  - La comunicazione comportamentale predefinita è consentita su un protocollo diverso da HTTPS
  - La comunicazione basata su comportamenti personalizzati è consentita su un protocollo diverso da HTTPS

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un indirizzo e-mail a cui desideri ricevere le notifiche di violazione

Limitazioni

- Questo controllo di sicurezza non verifica le distribuzioni Cloudfront esistenti a meno che non sia stato apportato un aggiornamento alla distribuzione.
- CloudFront è considerato un servizio globale e non è legato a una regione AWS specifica. Tuttavia, la registrazione delle API Amazon CloudWatch Logs e AWS Cloudtrail per i servizi globali avviene nella regione Stati Uniti orientali (Virginia settentrionale) (). `us-east-1` Pertanto, questo modulo di controllo di sicurezza CloudFront deve essere implementato e mantenuto in. `us-east-1` Questa singola implementazione monitora tutte le distribuzioni per. CloudFront Non distribuire il controllo di sicurezza in altre regioni AWS. (La distribuzione in altre regioni comporterà l'impossibilità di avviare CloudWatch Events e la funzione Lambda e l'assenza di notifiche SNS.)
- Questa soluzione è stata sottoposta a test approfonditi con le distribuzioni di contenuti CloudFront Web. Non copre le distribuzioni di streaming RTMP (Real-Time Messaging Protocol).

Architecture

Stack tecnologico Target

- funzione Lambda
- Argomento SNS
- EventBridge Regola Amazon

Architettura Target

Automazione e scalabilità

- Se utilizzi AWS Organizations, puoi utilizzare AWS Cloudformation StackSets per distribuire il modello allegato su più account che desideri monitorare.

Tools (Strumenti)

Servizi AWS

- AWS CloudFormation: CloudFormation è un servizio che ti aiuta a modellare e configurare le risorse AWS utilizzando l'infrastruttura come codice.
- Amazon EventBridge: EventBridge fornisce un flusso di dati in tempo reale dalle tue applicazioni, applicazioni SaaS (SaaS) e servizi AWS, indirizzando tali dati verso destinazioni come le funzioni Lambda.
- AWS Lambda — Lambda supporta l'esecuzione di codice senza effettuare il provisioning o la gestione di server.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti altamente scalabile che può essere utilizzato per un'ampia gamma di soluzioni di storage, tra cui siti Web, applicazioni mobili, backup e data lake.
- Amazon SNS: Amazon SNS coordina e gestisce la consegna o l'invio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail. I sottoscrittori ricevono tutti gli stessi messaggi pubblicati sugli argomenti ai quali sono hanno effettuato la sottoscrizione.

Codice

Il codice allegato include:

- Un file.zip che contiene il codice Lambda (index.py)
- Un CloudFormation modello (file.yml) che esegui per distribuire il codice Lambda

Epiche

Carica il controllo di sicurezza

Operazione	Description	Competenze richieste
Crea il bucket S3 per il codice Lambda.	Sulla console Amazon S3, crea un bucket S3 con un nome univoco che non contenga barre iniziali. Il nome di un bucket S3 è unico a livello globale e lo spazio dei nomi è condiviso da tutti gli account AWS. Il bucket S3	Architetto del cloud

Operazione	Description	Competenze richieste
	deve trovarsi nella regione in cui intendi distribuire il codice Lambda.	
Carica il codice Lambda nel bucket S3.	Carica il codice Lambda (file cloudfront_ssl_log_lambda.zip) fornito nella sezione Allegati nel bucket S3 che hai creato nel passaggio precedente.	Architetto del cloud

Implementa il modello CloudFormation

Operazione	Description	Competenze richieste
Implementa il CloudFormation modello.	Sulla CloudFormation console AWS, nella stessa regione AWS del bucket S3, distribui sci il CloudFormation modello (cloudfront-ssl-logging.yml) fornito nella sezione Allegati.	Architetto del cloud
Specificate il nome del bucket S3.	Per il parametro S3 Bucket, specifica il nome del bucket S3 che hai creato nella prima epic.	Architetto del cloud
Specificare il nome della chiave Amazon S3 per il file Lambda.	Per il parametro S3 Key, specifica la posizione Amazon S3 del file.zip del codice Lambda nel tuo bucket S3. Non includere barre iniziali (ad esempio, puoi inserire lambda.zip o controls/lambda.zip).	Architetto del cloud

Operazione	Description	Competenze richieste
Fornisci un indirizzo email di notifica.	Per il parametro e-mail di notifica, fornisci un indirizzo e-mail a cui desideri ricevere le notifiche di violazione.	Architetto del cloud
Definisci il livello di registrazione.	Per il parametro Lambda Logging level, definisci il livello di registrazione per la tua funzione Lambda. Seleziona uno dei seguenti valori: • INFO per ricevere messaggi informativi dettagliati sullo stato di avanzamento dell'applicazione. • ERRORE nell'ottenere informazioni sugli eventi di errore che potrebbero comunque consentire all'applicazione di continuare a funzionare. • AVVISO per ottenere informazioni su situazioni potenzialmente dannose.	Architetto del cloud

Confermare la sottoscrizione

Operazione	Description	Competenze richieste
Confermare la sottoscrizione.	Quando il CloudFormation modello è stato distribuito correttamente, viene creato un nuovo argomento SNS e viene inviato un messaggio	Architetto del cloud

Operazione	Description	Competenze richieste
	di iscrizione all'indirizzo e-mail fornito. È necessario confermare questa iscrizione e-mail per ricevere le notifiche di violazione.	

Risorse correlate

- CloudFormation Informazioni su AWS
- Creazione di uno stack sulla CloudFormation console AWS (CloudFormation documentazione)
- CloudFront registrazione (documentazione) CloudFront
- Informazioni su Amazon S3
- Informazioni su AWS Lambda

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Scegli un flusso di autenticazione Amazon Cognito per applicazioni aziendali

Michael Daehnert e Fabian Jahnke, Amazon Web Services

Riepilogo

Amazon Cognito fornisce autenticazione, autorizzazione e gestione degli utenti per applicazioni Web e mobili. Offre funzionalità vantaggiose per l'autenticazione delle identità federate. Per renderlo operativo, gli architetti tecnici devono decidere come utilizzare tali funzionalità.

Amazon Cognito supporta più flussi per le richieste di autenticazione. Questi flussi definiscono in che modo gli utenti possono verificare la propria identità. La decisione su quale flusso di autenticazione utilizzare dipende dai requisiti specifici dell'applicazione e può diventare complessa. Questo modello consente di decidere quale flusso di autenticazione è più adatto alla propria applicazione aziendale. Presuppone che tu abbia già una conoscenza di base di Amazon Cognito, OpenID Connect (OIDC) e della federazione e ti guida attraverso i dettagli sui diversi flussi di autenticazione federati.

Questa soluzione è destinata ai responsabili delle decisioni tecniche. Ti aiuta a comprendere i diversi flussi di autenticazione e a mapparli in base ai requisiti dell'applicazione. I responsabili tecnici dovrebbero raccogliere le informazioni necessarie per avviare le integrazioni di Amazon Cognito. Poiché le organizzazioni aziendali si concentrano principalmente sulla federazione SAML, questo modello include descrizioni per i pool di utenti di Amazon Cognito con federazione SAML.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Ruoli e autorizzazioni di AWS Identity and Access Management (IAM) con accesso completo ad Amazon Cognito
- (Facoltativo) Accesso al tuo provider di identità (IdP), come Microsoft Entra ID, Active Directory Federation Service (AD FS) o Okta
- Un elevato livello di esperienza per la tua applicazione
- Conoscenze di base di Amazon Cognito, OpenID Connect (OIDC) e federazione

Limitazioni

- Questo modello si concentra sui pool di utenti e sui provider di identità di Amazon Cognito. Per informazioni sui pool di identità di Amazon Cognito, consulta la sezione Informazioni aggiuntive.

Architecture

Utilizza la tabella seguente per aiutarti a scegliere un flusso di autenticazione. Ulteriori informazioni su ciascun flusso sono disponibili in questa sezione.

È necessaria machine-to-machine l'autenticazione?	La tua app è un'applicazione basata sul Web in cui il frontend viene renderizzato sul server?	La tua app è un'applicazione a pagina singola (SPA) o un'applicazione frontend basata su dispositivi mobili?	La tua applicazione richiede token di aggiornamento per la funzionalità «mantienimi connesso»?	Il frontend offre un meccanismo di reindirizzamento basato su browser?	Flusso consigliato di Amazon Cognito
Sì	No	No	No	No	Flusso delle credenziali del cliente
No	Sì	No	Sì	Sì	Flusso del codice di autorizzazione
No	No	Sì	Sì	Sì	Flusso del codice di autorizzazione con Proof Key for Code Exchange (PKCE)

No	No	No	No	No	Flusso di password del proprietario della risorsa*

* Il flusso della password del proprietario della risorsa deve essere utilizzato solo se assolutamente necessario. Per ulteriori informazioni, consulta la sezione relativa al flusso della password del proprietario della risorsa in questo modello.

Flusso delle credenziali del client

Il flusso Client Credentials è il più breve tra i flussi di Amazon Cognito. Dovrebbe essere usato se i sistemi o i servizi comunicano tra loro senza alcuna interazione da parte dell'utente. Il sistema richiedente utilizza l'ID client e il client secret per recuperare un token di accesso. Poiché entrambi i sistemi funzionano senza l'interazione dell'utente, non è richiesta alcuna fase di consenso aggiuntiva.

Il diagramma illustra quanto segue:

1. L'applicazione 1 invia una richiesta di autenticazione con l'ID client e il segreto del client all'endpoint Amazon Cognito e recupera un token di accesso.
2. L'Applicazione 1 utilizza questo token di accesso per ogni chiamata successiva all'Applicazione 2.
3. L'applicazione 2 convalida il token di accesso con Amazon Cognito.

Questo flusso deve essere utilizzato:

- Per comunicazioni tra applicazioni senza interazione da parte dell'utente

Questo flusso non deve essere usato:

- Per qualsiasi comunicazione in cui sono possibili interazioni con l'utente

Flusso del codice di autorizzazione

Il flusso del codice di autorizzazione è per l'autenticazione classica basata sul Web. In questo flusso, il backend gestisce tutto lo scambio e l'archiviazione dei token. Il client basato su browser non vede

i token effettivi. Questa soluzione viene utilizzata per applicazioni scritte in framework come .NET Core, Jakarta Faces o Jakarta Server Pages (JSP).

Il flusso del codice di autorizzazione è un flusso basato sul reindirizzamento. Il client deve essere in grado di interagire con il browser Web o un client simile. Il client viene reindirizzato a un server di autenticazione e si autentica su questo server. Se il client si autentica correttamente, viene reindirizzato nuovamente al server.

Il diagramma illustra quanto segue:

1. Il client invia una richiesta al server Web.
2. Il server Web reindirizza il client ad Amazon Cognito utilizzando un codice di stato HTTP 302. Il client segue automaticamente questo reindirizzamento all'accesso IdP configurato.
3. L'IdP verifica la presenza di una sessione del browser esistente sul lato IdP. Se non ne esiste nessuna, l'utente riceve una richiesta di autenticazione fornendo nome utente e password. L'IdP risponde con un token SAML ad Amazon Cognito.
4. Amazon Cognito restituisce il successo con un token web JSON (JWT), in particolare un token di codice. Il server Web chiama /oauth2/token per scambiare il token di codice con un token di accesso. Il server Web invia l'ID client e il segreto del client ad Amazon Cognito per la convalida.
5. Il token di accesso viene utilizzato per ogni chiamata successiva ad altre applicazioni.
6. Altre applicazioni convalidano il token di accesso con Amazon Cognito.

Questo flusso deve essere utilizzato:

- Se l'utente è in grado di interagire con il browser Web o il client. Il codice dell'applicazione viene eseguito e renderizzato sul server per garantire che nessun segreto venga esposto al browser.

Questo flusso non deve essere usato:

- Per applicazioni a pagina singola (SPAs) o app mobili perché sono renderizzate sul client e non devono utilizzare i segreti del client.

Flusso del codice di autorizzazione con PKCE

Il flusso del codice di autorizzazione con Proof Key for Code Exchange (PKCE) deve essere utilizzato per applicazioni a pagina singola e applicazioni mobili. È il successore del flusso implicito ed è più

sicuro perché utilizza PKCE. PKCE è un'estensione della concessione del codice di autorizzazione OAuth 2.0 per i clienti pubblici. PKCE protegge dal riscatto dei codici di autorizzazione intercettati.

Il diagramma illustra quanto segue:

1. L'applicazione crea un verificatore di codice e una verifica del codice. Si tratta di valori unici e ben definiti che vengono inviati ad Amazon Cognito per riferimenti futuri.
2. L'applicazione chiama l'endpoint /oauth2/authorization di Amazon Cognito. Reindirizza automaticamente l'utente all'accesso IdP configurato.
3. L'IdP verifica la presenza di una sessione esistente. Se non ne esiste nessuna, l'utente riceve una richiesta di autenticazione fornendo nome utente e password. L'IdP risponde con un token SAML ad Amazon Cognito.
4. Dopo che Amazon Cognito ha restituito il successo con un token di codice, il server Web chiama / oauth2/token per scambiare il token di codice con un token di accesso.
5. Il token di accesso viene utilizzato per ogni chiamata successiva ad altre applicazioni.
6. Le altre applicazioni convalidano il token di accesso con Amazon Cognito.

Questo flusso deve essere utilizzato:

- Per SPAs le nostre applicazioni mobili

Questo flusso non deve essere usato:

- Se il backend dell'applicazione gestisce l'autenticazione

Flusso della password del proprietario della risorsa

Il flusso Resource Owner Password è destinato alle applicazioni senza funzionalità di reindirizzamento. Viene creato creando un modulo di accesso nella propria applicazione. L'accesso viene verificato su Amazon Cognito tramite una chiamata CLI o SDK anziché affidarsi ai flussi di reindirizzamento. La federazione non è possibile in questo flusso di autenticazione perché la federazione richiede reindirizzamenti basati su browser.

Il diagramma illustra quanto segue:

1. L'utente inserisce le proprie credenziali in un modulo di accesso fornito dall'applicazione.
2. L'AWS Command Line Interface (AWS CLI) effettua admin-initiated-authuna chiamata ad Amazon Cognito.

> **Note**
>
> In alternativa, puoi utilizzare AWS SDKs al posto dell'interfaccia a riga di comando di AWS.

3. Amazon Cognito restituisce un token di accesso.
4. Il token di accesso viene utilizzato per ogni chiamata successiva ad altre applicazioni.
5. Le altre applicazioni convalidano il token di accesso con Amazon Cognito.

Questo flusso deve essere utilizzato:

- Durante la migrazione dei client esistenti che utilizzano la logica di autenticazione diretta (come l'autenticazione di accesso di base o l'autenticazione con accesso digest) convertendo le credenziali archiviate in un token di accesso OAuth

Questo flusso non deve essere usato:

- Se desideri utilizzare identità federate
- Se l'applicazione supporta i reindirizzamenti

Tools (Strumenti)

Servizi AWS

- Amazon Cognito fornisce autenticazione, autorizzazione e gestione degli utenti per app Web e mobili.

Altri strumenti

- Il debugger JSON web token (JWT) è uno strumento di convalida JWT basato sul web.

Epiche

Valuta la tua candidatura

Operazione	Description	Competenze richieste
Definire i requisiti di autentica zione.	Valuta la tua applicazione in base ai tuoi requisiti di autenticazione specifici.	Sviluppatore di app, architetto dell'app
Allinea i requisiti ai flussi di autenticazione.	Nella sezione Architett ura, utilizza la tabella delle decisioni e le spiegazioni di ogni flusso per scegliere il flusso di autenticazione di Amazon Cognito.	Sviluppatore di app, General AWS, Architetto dell'app

Configura il pool di utenti di Amazon Cognito

Operazione	Description	Competenze richieste
Crea un pool di utenti.	1. Accedi alla Console di gestione AWS, quindi apri la console Amazon Cognito. 2. Crea un nuovo pool di utenti Cognito. Per istruzioni, consulta i pool di utenti di Amazon Cognito. 3. Aggiorna le impostazioni e gli attributi del pool di utenti secondo necessità. Ad esempio, imposta una politica di password per il pool di utenti. Non create ancora client per le app.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
(Facoltativo) Configura un provider di identità.	1. Crea un provider di identità SAML nel pool di utenti di Amazon Cognito. Per istruzioni, consulta Aggiungere e gestire provider di identità SAML in un pool di utenti. 2. Configura il tuo provider di identità SAML di terze parti per utilizzare la federazione per i pool di utenti di Amazon Cognito. Per ulteriori informazioni, consulta Configurazione del provider di identità SAML di terze parti. Se utilizzi AD FS, consulta Creazione di una federazione AD FS per la tua app Web utilizzando i pool di utenti di Amazon Cognito (post sul blog AWS).	General AWS, amministratore della federazione

Operazione	Description	Competenze richieste
Crea un client per l'app.	1. Crea un client di app per il pool di utenti. Per istruzioni, consulta Creazione di un client per l'app. Tenere presente quanto segue: • Modifica le impostazioni secondo necessità, ad esempio le scadenze dei token. • Se il flusso di autenticazione non richiede un segreto client, deselezionare la casella di controllo Genera segreto del client. 2. Scegli le impostazioni del client dell'app per modificarne l'integrazione con un accesso al pool di utenti (nome utente e password) o un accesso federato tramite un IdP basato su SAML. 3. Abilita il tuo IdP definendo URLs e definendo OAuth flussi o ambiti in base alle esigenze.	Informazioni generali su AWS

Integra l'applicazione con Amazon Cognito

Operazione	Description	Competenze richieste
Dettagli sull'integrazione con Amazon Cognito di Exchange.	A seconda del flusso di autenticazione, condividi le informazioni di Amazon	Sviluppatore di app, General AWS

Operazione	Description	Competenze richieste
	Cognito con l'applicazione, come l'ID del pool di utenti e l'ID client dell'app.	
Implementa l'autenticazione Amazon Cognito.	Dipende dal flusso di autenticazione scelto, dal linguaggio di programmazione e dai framework che utilizzi. Per alcuni collegamenti per iniziare, consulta la sezione Risorse correlate.	Sviluppatore di app

Risorse correlate

Documentazione AWS

- Flusso di autenticazione del pool di utenti
- Verifica di un token web JSON
- Accedi ai servizi AWS da un'app ASP.NET Core utilizzando i pool di identità di Amazon Cognito
- Framework e: SDKs
  - Autenticazione Amazon Amplify
  - Esempi di Amazon Cognito Identity Provider (documentazione AWS SDK per Java 2.x)
  - Autenticazione degli utenti con Amazon Cognito (documentazione dell'SDK AWS per .NET)

Post sul blog di AWS

- Authorization @Edge utilizzando i cookie: proteggi i tuoi CloudFront contenuti Amazon dal download da parte di utenti non autenticati
- Creazione di una federazione AD FS per la tua app Web utilizzando i pool di utenti di Amazon Cognito

Partner di implementazione

- Partner AWS per soluzioni di autenticazione

Informazioni aggiuntive

DOMANDE FREQUENTI

Perché il flusso implicito è obsoleto?

Dal rilascio del framework OAuth 2.1, il flusso Implicit è contrassegnato come obsoleto per motivi di sicurezza. In alternativa, utilizzate il flusso del codice di autorizzazione con PKCE descritto nella sezione Architettura.

Cosa succede se Amazon Cognito non offre alcune funzionalità di cui ho bisogno?

I partner AWS offrono diverse integrazioni per soluzioni di autenticazione e autorizzazione. Per ulteriori informazioni, consulta AWS Partners for authentication solutions.

Che dire dei flussi del pool di identità di Amazon Cognito?

I pool di utenti e le identità federate di Amazon Cognito servono per l'autenticazione. I pool di identità di Amazon Cognito vengono utilizzati per l'autorizzazione dell'accesso alle risorse AWS richiedendo credenziali AWS temporanee. Lo scambio di token ID e token di accesso per i pool di identità non viene discusso in questo schema. Per ulteriori informazioni, consulta Qual è la differenza tra i pool di utenti e i pool di identità di Amazon Cognito e gli scenari comuni di Amazon Cognito.

Fasi successive

Questo modello fornisce una panoramica dei flussi di autenticazione di Amazon Cognito. Come passo successivo, è necessario scegliere l'implementazione dettagliata per il linguaggio di programmazione dell'applicazione. Offerta SDKs e framework in più lingue, utilizzabili con Amazon Cognito. Per riferimenti utili, consulta la sezione Risorse correlate.

Crea regole AWS Config personalizzate utilizzando AWS CloudFormation Guard le politiche

Andrew Lok, Nicole Brown, Kailash Havildar e Tanya Howell, Amazon Web Services

Riepilogo

AWS Configle regole ti aiutano a valutare le tue risorse e il loro stato di configurazione di destinazione. AWS Esistono due tipi di AWS Config regole: gestite e personalizzate. È possibile creare regole personalizzate con AWS Lambda funzioni o con AWS CloudFormation Guard(GitHub), un policy-as-code linguaggio.

Le regole create con Guard forniscono un controllo più granulare rispetto alle regole gestite e in genere sono più facili da configurare rispetto alle regole Lambda completamente personalizzate. Questo approccio offre a ingegneri e architetti la possibilità di creare regole senza dover conoscere Python, NodeJS o Java, necessari per implementare regole personalizzate tramite Lambda.

Questo modello fornisce modelli utilizzabili, esempi di codice e approcci di implementazione per aiutarti ad adottare regole personalizzate con Guard. Utilizzando questo modello, un amministratore può utilizzarlo AWS Config per creare regole di conformità personalizzate con attributi degli elementi di configurazione. Ad esempio, gli sviluppatori possono utilizzare le policy di Guard rispetto agli elementi di AWS Config configurazione per monitorare continuamente lo stato delle AWS risorse distribuite AWS e non, rilevare le violazioni delle regole e avviare automaticamente la correzione.

Obiettivi

Dopo aver letto questo schema, dovresti essere in grado di:

- Scopri come il codice di policy di Guard interagisce con il AWS Config servizio.
- Implementa lo Scenario 1, che è una regola AWS Config personalizzata che utilizza la sintassi Guard per convalidare la conformità per i volumi crittografati. Questa regola verifica che l'unità sia in uso e verifica che il tipo di unità sia gp3.
- Implementa lo Scenario 2, una regola AWS Config personalizzata che utilizza la sintassi Guard per convalidare la conformità di Amazon GuardDuty . Questa regola verifica che GuardDuty i registratori abbiano la protezione Amazon Simple Storage Service (Amazon S3) e la protezione Amazon Elastic Kubernetes Service (Amazon EKS) abilitate.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- AWS Config, configuralo nel tuo Account AWS

Limitazioni

- Le regole personalizzate di Guard sono in grado di interrogare solo le coppie chiave-valore in un record JSON dell'elemento di configurazione di destinazione

Architecture

Si applica la sintassi Guard a una AWS Config regola come politica personalizzata. AWS Config acquisisce il codice JSON gerarchico di ciascuna delle risorse specificate. Il JSON dell'elemento di AWS Config configurazione contiene coppie chiave-valore. Questi attributi vengono utilizzati nella sintassi Guard come variabili assegnate al valore corrispondente.

Di seguito è riportata una spiegazione della sintassi Guard. Le variabili dell'elemento di configurazione JSON vengono utilizzate e precedute da un carattere. `%`

```
# declare variable
let <variable name> = <'value'>

# create rule and assign condition and policy
    rule <rule name> when
        <CI json key> == <"CI json value"> {
            <top level CI json key>.<next level CI json key> == %<variable name>
        }
```

Scenario 1: volumi Amazon EBS

Lo scenario 1 implementa una regola AWS Config personalizzata che utilizza la sintassi Guard per convalidare la conformità dei volumi crittografati. Questa regola verifica che l'unità sia in uso e verifica che il tipo di unità sia gp3.

Di seguito è riportato un esempio di elemento di AWS Config configurazione per lo scenario 1. In questo elemento di configurazione sono presenti tre coppie chiave-valore utilizzate come variabili

nella policy Guard:`volumestatus`,`volumeencryptionstatus`, e. `volumetype` Inoltre, la `resourceType` chiave viene utilizzata come filtro nella policy Guard.


```
{
  "version": "1.3",
  "accountId": "111111111111",
  "configurationItemCaptureTime": "2023-01-15T19:04:45.402Z",
  "configurationItemStatus": "ResourceDiscovered",
  "configurationStateId": "4444444444444",
  "configurationItemMD5Hash": "",
  "arn": "arn:aws:ec2:us-west-2:111111111111:volume/vol-222222222222",
  "resourceType": "AWS::EC2::Volume",
  "resourceId": "vol-222222222222",
  "awsRegion": "us-west-2",
  "availabilityZone": "us-west-2b",
  "resourceCreationTime": "2023-01-15T19:03:22.247Z",
  "tags": {},
  "relatedEvents": [],
  "relationships": [
    {
      "resourceType": "AWS::EC2::Instance",
      "resourceId": "i-33333333333333333",
      "relationshipName": "Is attached to Instance"
    }
  ],
  "configuration": {
    "attachments": [
      {
        "attachTime": "2023-01-15T19:03:22.000Z",
        "device": "/dev/xvda",
        "instanceId": "i-33333333333333333",
        "state": "attached",
        "volumeId": "vol-222222222222",
        "deleteOnTermination": true,
        "associatedResource": null,
        "instanceOwningService": null
      }
    ],
    "availabilityZone": "us-west-2b",
    "createTime": "2023-01-15T19:03:22.247Z",
    "encrypted": false,
    "kmsKeyId": null,
    "outpostArn": null,
```

```
    "size": 8,
    "snapshotId": "snap-55555555555555555",
    "state": "in-use",
    "volumeId": "vol-222222222222",
    "iops": 100,
    "tags": [],
    "volumeType": "gp2",
    "fastRestored": null,
    "multiAttachEnabled": false,
    "throughput": null,
    "sseType": null
  },
  "supplementaryConfiguration": {}
}
```

Di seguito è riportato un esempio di utilizzo della sintassi Guard per definire le variabili e le regole nello scenario 1. Nel seguente esempio:

- Le prime tre righe definiscono le variabili utilizzando il `let` comando. A esse viene assegnato un nome e un valore derivati dagli attributi dell'elemento di configurazione.
- Il blocco di `compliancecheck` regole aggiunge una dipendenza condizionale quando cerca una coppia `resourceType` chiave-valore corrispondente. `AWS::EC2::Volume` Se viene trovata una corrispondenza, la regola passa attraverso il resto degli attributi JSON e cerca le corrispondenze nelle tre condizioni seguenti:,, e. `state` `encrypted` `volumeType`

```
let volumestatus = 'available'
let volumetype = 'gp3'
let volumeencryptionstatus = true

    rule compliancecheck when
        resourceType == "AWS::EC2::Volume" {
            configuration.state == %volumestatus
            configuration.encrypted == %volumeencryptionstatus
            configuration.volumeType == %volumetype
        }
```

Per la politica personalizzata completa di Guard che implementa questa regola personalizzata, consulta awsconfig-guard-cft.yaml o awsconfig-guard-tf-ec 2vol.json nel repository del codice. GitHub Per il codice HashiCorp Terraform che implementa questa politica personalizzata in Guard, consulta .json nel repository del codice. awsconfig-guard-tf-example

GuardDuty Scenario 2: conformità

Lo scenario 2 implementa una regola AWS Config personalizzata che utilizza la sintassi Guard per convalidare la conformità di Amazon GuardDuty . Questa regola verifica che nei GuardDuty registratori siano abilitati Amazon S3 Protection e Amazon EKS Protection. Verifica inoltre che i GuardDuty risultati vengano pubblicati ogni 15 minuti. Questo scenario potrebbe essere implementato in tutti Account AWS e Regioni AWS in un'organizzazione (in AWS Organizations).

Di seguito è riportato un esempio di elemento di AWS Config configurazione per lo scenario 2. In questo elemento di configurazione sono presenti tre coppie chiave-valore utilizzate come variabili nella policy Guard:`FindingPublishingFrequency`,`S3Logs`, e. `Kubernetes` Inoltre, la `resourceType` chiave viene utilizzata come filtro nella politica.

```
{
  "version": "1.3",
  "accountId": "111111111111",
  "configurationItemCaptureTime": "2023-11-27T13:34:28.888Z",
  "configurationItemStatus": "OK",
  "configurationStateId": "7777777777777",
  "configurationItemMD5Hash": "",
  "arn": "arn:aws:guardduty:us-
west-2:111111111111:detector/66666666666666666666666666666666",
  "resourceType": "AWS::GuardDuty::Detector",
  "resourceId": "66666666666666666666666666666666",
  "resourceName": "66666666666666666666666666666666",
  "awsRegion": "us-west-2",
  "availabilityZone": "Regional",
  "resourceCreationTime": "2020-02-17T02:48:04.511Z",
  "tags": {},
  "relatedEvents": [],
  "relationships": [],
  "configuration": {
    "Enable": true,
    "FindingPublishingFrequency": "FIFTEEN_MINUTES",
    "DataSources": {
      "S3Logs": {
        "Enable": true
      },
      "Kubernetes": {
        "AuditLogs": {
          "Enable": true
        }
```

```
      }
    },

    "Id": "66666666666666666666666666666666",
    "Tags": []
  },
  "supplementaryConfiguration": {
    "CreatedAt": "2020-02-17T02:48:04.511Z"
  }
}
```

Di seguito è riportato un esempio di utilizzo della sintassi Guard per definire le variabili e le regole nello scenario 2. Nel seguente esempio:

- Le prime tre righe definiscono le variabili utilizzando il `let` comando. A esse viene assegnato un nome e un valore derivati dagli attributi dell'elemento di configurazione.
- Il blocco di `compliancecheck` regole aggiunge una dipendenza condizionale quando cerca una coppia `resourceType` chiave-valore corrispondente. `AWS::GuardDuty::Detector` Se viene trovata una corrispondenza, la regola passa attraverso il resto degli attributi JSON e cerca le corrispondenze nelle tre condizioni seguenti:,, e. `S3Logs.Enable` `Kubernetes.AuditLogs.Enable` `FindingPublishingFrequency`

```
let s3protection = true
let kubernetesprotection = true
let publishfrequency = 'FIFTEEN_MINUTES'

    rule compliancecheck when
        resourceType == "AWS::GuardDuty::Detector" {
            configuration.DataSources.S3Logs.Enable == %s3protection
            configuration.DataSources.Kubernetes.AuditLogs.Enable ==
 %kubernetesprotection
            configuration.FindingPublishingFrequency == %publishfrequency
        }
```

Per la politica personalizzata completa di Guard che implementa questa regola personalizzata, consulta awsconfig-guard-cft-gd.yaml nel repository del codice. GitHub Per il codice HashiCorp Terraform che implementa questa politica personalizzata in Guard, consulta .json nel repository del codice. awsconfig-guard-tf-gd

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante il loro ciclo di vita in tutte le regioni. Account AWS
- AWS Configfornisce una visione dettagliata delle risorse disponibili Account AWS e di come sono configurate. Ti aiuta a identificare in che modo le risorse sono correlate tra loro e in che modo le loro configurazioni sono cambiate nel tempo.

Altri strumenti

- HashiCorp Terraform è uno strumento Infrastructure as Code (IaC) che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud.

Archivio di codici

Il codice per questo pattern è disponibile nel AWS CloudFormation Guard repository GitHub AWS Config with. Questo archivio di codice contiene esempi per entrambi gli scenari descritti in questo modello.

Epiche

Creazione di AWS Config regole personalizzate

Operazione	Description	Competenze richieste
(Facoltativo) Seleziona le coppie chiave-valore per la regola.	Completa questi passaggi se stai definendo una politica Guard personalizzata. Se stai utilizzando una delle politiche di esempio per lo scenario 1 o 2, salta questi passaggi. 1. Accedi a Console di gestione AWS e apri la AWS Config console.	Amministratore AWS, ingegnere della sicurezza

Operazione	Description	Competenze richieste
	2. Nella barra di navigazione a sinistra, scegli Risorse. 3. Nell'inventario delle risorse, scegli il tipo di risorsa per cui desideri creare una regola AWS Config personalizzata. 4. Seleziona Visualizza dettagli. 5. Scegli Visualizza elemento di configurazione (JSON). Questa sezione si espande per mostrare l'elemento di configurazione in formato JSON. 6. Identifica le coppie chiave-valore per le quali desideri creare una AWS Config regola personalizzata.	
Crea la regola personalizzata.	Utilizzando le coppie chiave-valore identificate in precedenza o utilizzando una delle politiche Guard di esempio fornite, segui le istruzioni in Creazione di regole di policy AWS Config personalizzate per creare una regola personalizzata.	Amministratore AWS, ingegnere della sicurezza

Operazione	Description	Competenze richieste
Convalida la regola personalizzata.	Effettua una delle seguenti operazioni per convalidare la regola Guard personalizzata: • Immettete il seguente comando in AWS Command Line Interface (AWS CLI). `cfn-guard validate -r guard-s3.guard -d s3bucket-prod-pass.json` • Segui le istruzioni in modalità Detective in Evaluating Your Resources with AWS Config Rules per implementare la regola. AWS Config Verifica che la sintassi di Guard corrispondente correttamente alle risorse corrispondenti nell'account o nel file di destinazione.	Amministratore AWS, ingegnere della sicurezza

risoluzione dei problemi

Problema	Soluzione
Prova la policy di Guard al di fuori di AWS Config	I test unitari possono essere eseguiti sul dispositivo locale o in un ambiente di sviluppo integrato (IDE), come un AWS Cloud9 IDE. Per eseguire il test unitario, procedi come segue: 1. Installa la AWS CloudFormation Guard CLI e le sue dipendenze.

Problema	Soluzione

	2. Salva un esempio di CI in formato JSON sulla tua workstation come file.json.
	3. Salva la GuardDuty policy sulla tua workstation come file.guard.
	4. Nella CLI Guard, inserisci il seguente comando per convalidare il file JSON di esempio utilizzando la policy Guard.
	`cfn-guard validate \`
	`-r guard-s3.guard \`
	`-d s3bucket-prod-pass.json`

Esegui il debug di una regola personalizzata AWS Config	Nella tua politica Guard, modifica il `EnableDebugLogDelivery` valore in `true`. Il valore predefinito è `false`. I messaggi di registro vengono archiviati in Amazon CloudWatch.

Risorse correlate

AWS documentazione

- Creazione di regole di policy AWS Config personalizzate (AWS Config documentazione)
- AWS CloudFormation Guard Regole di scrittura (documentazione di Guard)

AWS post di blog e workshop

- Presentazione della AWS CloudFormation Guard versione 2.0 (post AWS sul blog)

Altre risorse

- AWS CloudFormation Guard (GitHub)
- AWS CloudFormation Guard Documentazione CLI () GitHub

Crea un report consolidato dei risultati di sicurezza di Prowler da più Account AWS

Mike Virgilio, Jay Durga e Andrea Di Fabio, Amazon Web Services

Riepilogo

Prowler (GitHub) è uno strumento da riga di comando open source che può aiutarti a valutare, controllare e monitorare i tuoi account Amazon Web Services (AWS) per verificare la conformità alle migliori pratiche di sicurezza. In questo modello, si implementa Prowler in un sistema centralizzato Account AWS dell'organizzazione, gestito da AWS Organizations e quindi si utilizza Prowler per eseguire una valutazione della sicurezza di tutti gli account dell'organizzazione.

Sebbene esistano molti metodi per implementare e utilizzare Prowler per una valutazione, questa soluzione è stata progettata per un'implementazione rapida, l'analisi completa di tutti gli account dell'organizzazione o degli account target definiti e la rendicontazione accessibile dei risultati di sicurezza. In questa soluzione, quando Prowler completa la valutazione della sicurezza di tutti gli account dell'organizzazione, consolida i risultati. Inoltre, filtra tutti i messaggi di errore previsti, come gli errori relativi alle restrizioni che impediscono a Prowler di scansionare i bucket Amazon Simple Storage Service (Amazon S3) negli account forniti tramite. AWS Control Tower I risultati filtrati e consolidati vengono riportati in un modello di Microsoft Excel incluso in questo modello. È possibile utilizzare questo rapporto per identificare potenziali miglioramenti per i controlli di sicurezza nella propria organizzazione.

Questa soluzione è stata progettata tenendo presente quanto segue:

- I AWS CloudFormation modelli riducono lo sforzo richiesto per distribuire le AWS risorse secondo questo schema.
- È possibile modificare i parametri nei CloudFormation modelli e nello script prowler_scan.sh al momento della distribuzione per personalizzare i modelli per l'ambiente.
- Le velocità di valutazione e reporting di Prowler sono ottimizzate attraverso l'elaborazione parallela, i risultati aggregati Account AWS, la reportistica consolidata con le correzioni consigliate e le visualizzazioni generate automaticamente.
- L'utente non deve monitorare l'avanzamento della scansione. Una volta completata la valutazione, l'utente riceve una notifica tramite un argomento di Amazon Simple Notification Service (Amazon SNS) in modo che possa recuperare il report.

- Il modello di report ti aiuta a leggere e valutare solo i risultati pertinenti per l'intera organizzazione.

Prerequisiti e limitazioni

Prerequisiti

- E Account AWS per ospitare servizi e strumenti di sicurezza, gestiti come account membro di un'organizzazione in AWS Organizations. In questo schema, questo account viene definito account di sicurezza.
- Nell'account di sicurezza, è necessario disporre di una sottorete privata con accesso a Internet in uscita. Per istruzioni, consulta VPC con server in sottoreti private e NAT nella documentazione di Amazon Virtual Private Cloud (Amazon VPC). Puoi stabilire l'accesso a Internet utilizzando un gateway NAT fornito in una sottorete pubblica.
- Accesso all'account di AWS Organizations gestione o a un account con autorizzazioni di amministratore delegate per. CloudFormation Per istruzioni, consulta Registrare un amministratore delegato nella documentazione. CloudFormation
- Abilita l'accesso affidabile tra AWS Organizations e CloudFormation. Per istruzioni, consulta Abilita l'accesso affidabile con AWS Organizations nella CloudFormation documentazione.

Limitazioni

- L'obiettivo Account AWS deve essere gestito come organizzazione in AWS Organizations. Se non lo utilizzi AWS Organizations, puoi aggiornare il CloudFormation modello IAM-ProwlerExecRole .yaml e lo script prowler_scan.sh per il tuo ambiente. Fornisci invece un elenco delle regioni Account AWS IDs e delle regioni in cui desideri eseguire lo script.
- Il CloudFormation modello è progettato per distribuire l'istanza Amazon Elastic Compute Cloud EC2 (Amazon) in una sottorete privata con accesso a Internet in uscita. L' AWS Systems Manager agente (agente SSM) richiede l'accesso in uscita per raggiungere l'endpoint del AWS Systems Manager servizio e l'accesso in uscita è necessario per clonare l'archivio di codice e installare le dipendenze. Se si desidera utilizzare una sottorete pubblica, è necessario modificare il modello prowler-resources.yaml per associare un indirizzo IP elastico all'istanza. EC2

Versioni del prodotto

- Prowler versione 4.0 o successiva

Architecture

Il diagramma mostra il seguente processo:

1. Utilizzando Session Manager, una funzionalità di AWS Systems Manager, l'utente si autentica sull' EC2 istanza ed esegue lo script prowler_scan.sh. Questo script di shell esegue i passaggi da 2 a 8.
2. L' EC2 istanza assume il ruolo `ProwlerEC2Role` IAM, che concede le autorizzazioni per accedere al bucket S3 e per assumere i ruoli `ProwlerExecRole` IAM negli altri account dell'organizzazione.
3. L' EC2 istanza assume il ruolo `ProwlerExecRole` IAM nell'account di gestione dell'organizzazione e genera un elenco degli account dell'organizzazione.
4. L' EC2 istanza assume il ruolo `ProwlerExecRole` IAM negli account dei membri dell'organizzazione (denominati account di carico di lavoro nel diagramma dell'architettura) ed esegue una valutazione della sicurezza in ciascun account. I risultati vengono archiviati come file CSV e HTML sull'istanza. EC2

   > **Note**
   >
   > I file HTML sono un output della valutazione Prowler. A causa della natura dell'HTML, non vengono concatenati, elaborati o utilizzati direttamente in questo modello. Tuttavia, potrebbero essere utili per la revisione dei report sui singoli account.

5. L' EC2 istanza elabora tutti i file CSV per rimuovere gli errori noti e previsti e consolida i risultati rimanenti in un unico file CSV.
6. L' EC2 istanza racchiude i risultati dei singoli account e i risultati aggregati in un file zip.
7. L' EC2 istanza carica il file zip nel bucket S3.
8. Una EventBridge regola rileva il caricamento del file e utilizza un argomento Amazon SNS per inviare un'e-mail all'utente per informarlo del completamento della valutazione.
9. L'utente scarica il file zip dal bucket S3. L'utente importa i risultati nel modello di Excel e li esamina.

Tools (Strumenti)

Servizi AWS

- Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di elaborazione scalabile in. Cloud AWS Puoi avviare tutti i server virtuali di cui hai bisogno e dimensionarli rapidamente.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altro modo. Account AWS
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Organizationsè un servizio di gestione degli account che ti aiuta a consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Systems Managerti aiuta a gestire le applicazioni e l'infrastruttura in esecuzione in. Cloud AWS Semplifica la gestione delle applicazioni e delle risorse, riduce i tempi di rilevamento e risoluzione dei problemi operativi e aiuta a gestire le AWS risorse in modo sicuro su larga scala. Questo modello utilizza Session Manager, una funzionalità di Systems Manager.

Altri strumenti

- Prowler è uno strumento a riga di comando open source che ti aiuta a valutare, controllare e monitorare i tuoi account per quanto riguarda la conformità alle migliori pratiche di sicurezza e ad altri AWS framework e standard di sicurezza.

Archivio di codice

Il codice di questo modello è disponibile nel GitHub Multi-Account Security Assessment tramite il repository Prowler. L'archivio del codice contiene i seguenti file:

- prowler_scan.sh — Questo script bash viene utilizzato per avviare una valutazione della sicurezza di Prowler multipla, Account AWS in parallelo. Come definito in Prowler-Resources.yaml CloudFormationtemplate, questo script viene distribuito automaticamente nella cartella sull'istanza. `usr/local/prowler` EC2
- Prowler-resources.yaml: questo modello viene utilizzato per creare uno stack nell'account di sicurezza dell'organizzazione. CloudFormation Questo modello distribuisce tutte le risorse

necessarie per questo account per supportare la soluzione. Questo stack deve essere distribuito prima del modello IAM- ProwlerExecRole .yaml. Non è consigliabile distribuire queste risorse in un account che ospita carichi di lavoro di produzione critici.

> **Note**
>
> Se questo stack viene eliminato e ridistribuito, devi ricostruire il set di `ProwlerExecRole` stack per ricostruire le dipendenze tra account tra i ruoli IAM.

- IAM- ProwlerExecRole .yaml: utilizzi questo CloudFormation modello per creare un set di stack che distribuisce il ruolo `ProwlerExecRole` IAM in tutti gli account dell'organizzazione, incluso l'account di gestione.
- prowler-report-template.xlsm: si utilizza questo modello di Excel per elaborare i risultati di Prowler. Le tabelle pivot del rapporto forniscono funzionalità di ricerca, grafici e risultati consolidati.

Epiche

Preparati per l'implementazione

Operazione	Description	Competenze richieste
Clona il repository del codice.	1. In un'interfaccia a riga di comando, modificate la directory di lavoro nella posizione in cui desiderate archiviare i file di esempio. 2. Immetti il comando seguente: `git clone https://github.com/aws-samples/multi-account-security-assessment-via-prowler.git`	AWS DevOps
Esamina i modelli.	1. Nel repository clonato, aprite i file Prowler-	AWS DevOps

Operazione	Description	Competenze richieste
	Resources.yaml e IAM- .yaml. ProwlerEx ecRole 2. Esamina le risorse create da questi modelli e modifica i modelli in base alle esigenze del tuo ambiente. Per ulteriori informazioni, consulta Lavorare con i modelli nella CloudForm ation documentazione. 3. Salvate e chiudete i file Prowler-Resources.yaml e IAM- .yaml. ProwlerEx ecRole	

Crea gli CloudFormation stack

Operazione	Description	Competenze richieste
Fornisci risorse nell'account di sicurezza.	Utilizzando il modello prowler-resources.yaml, crei uno CloudFormation stack che distribuisce tutte le risorse richieste nell'account di sicurezza. Per istruzion i, consulta Creazione di uno stack nella documenta zione. CloudFormation Tieni presente quanto segue durante la distribuzione di questo modello:	AWS DevOps

Operazione	Description	Competenze richieste
	1. Nella pagina Specifica re il modello, scegliete Il modello è pronto, quindi caricate il file prowler-r esources.yaml. 2. Nella pagina Specificare i dettagli dello stack, nella casella Nome dello stack, immettere. `Prowler-R esources` 3. Nella sezione Parametri, inserisci quanto segue: • `VPCId`— Seleziona un VPC nell'account. • `SubnetId`— Seleziona una sottorete privata con accesso a Internet. Nota: se si seleziona una sottorete pubblica, all' EC2 istanza non verrà assegnato un indirizzo IP pubblico perché il CloudFormation modello, per impostazi one predefinita, non fornisce e non collega un indirizzo IP elastico. • `InstanceType` — Seleziona la dimension e dell'istanza in base al numero di valutazioni parallele:	

Operazione	Description	Competenze richieste
	• Per 10, sceglir6i.large . • Per 12, sceglir6i.xlarge . • Per 14-18 anni, scegli. r6i.2xlarge • InstanceImageId — Lascia l'impostazione predefinita per Amazon Linux. • KeyPairName — Se utilizzi SSH per l'accesso , specifica il nome di una coppia di key pair esistente. • PermittedSSHInbound — Se utilizzi SSH per l'accesso, specifica un blocco CIDR consentito. Se non utilizzi SSH, mantieni il valore predefinito di. 127.0.0.1 • BucketName — Il valore predefinito èprowler-output-<accountID>-<region> . È possibile modificarlo in base alle esigenze. Se si specifica un valore personalizzato, l'ID dell'account e la regione vengono aggiunti	

Operazione	Description	Competenze richieste
	automaticamente al valore specificato. • `EmailAddress` — Specificare un indirizzo e-mail per una notifica Amazon SNS quando Prowler completa la valutazione e carica il file.zip nel bucket S3. Nota: la configurazione dell'abbonamento SNS deve essere confermat a prima che Prowler completi la valutazio ne, altrimenti non verrà inviata alcuna notifica. • `IAMProwle rEC2Role` — Mantieni l'impostazione predefinita a meno che le convenzio ni di denominazione non richiedano un nome diverso per questo ruolo IAM. • `IAMProwlerExecRole` — Mantieni il valore predefinito a meno che non venga utilizzato un altro nome durante la distribuzione del file IAM-ProwlerExecRole .yaml. • `Parallelism` — Specificare il numero	

Operazione	Description	Competenze richieste
	di valutazioni parallele da eseguire. Assicurat evi che il valore nel `InstanceType` parametro supporti questo numero di valutazioni parallele. • `FindingOutput` — Se desideri escludere i risultati del pass, selezionaFail0nly. Ciò riduce notevolmente le dimensioni dell'output e si concentra sui controlli che potrebbero dover essere risolti. Se desideri includere i risultati del pass, seleziona `FailAndPass` . 4. Nella pagina Revisione , seleziona Le seguenti risorse richiedono funzional ità: [AWS::IAM::Role], quindi scegli Crea stack. 5. Dopo che lo stack è stato creato correttamente, nella CloudFormation console, nella scheda Outputs, copia l'`ProwlerEC 2Role` Amazon Resource Name (ARN). Questo ARN verrà utilizzato in un secondo momento durante	

Operazione	Description	Competenze richieste
	la distribuzione del file IAM-ProwlerExecRole .yaml.	

Operazione	Description	Competenze richieste
Fornisci il ruolo IAM negli account dei membri.	Nell'account di AWS Organizations gestione o in un account con autorizzazioni di amministratore delegato per CloudFormation, utilizza il modello IAM-ProwlerExecRole .yaml per creare un set di stack. CloudFormation Lo stack set implementa il ruolo `ProwlerExecRole` IAM in tutti gli account membri dell'organizzazione. Per istruzioni, consulta Creare un set di stack con autorizzazioni gestite dal servizio nella documentazione. CloudFormation Tieni presente quanto segue durante la distribuzione di questo modello: 1. In Prepara modello, scegli Il modello è pronto, quindi carica il file IAM- ProwlerExecRole .yaml. 2. Nella pagina Specificare StackSet i dettagli, assegnate un nome al set di stack. `IAM-ProwlerExecRole` 3. Nella sezione Parametri, inserisci quanto segue: • `AuthorizedARN` — Inserisci l'`ProwlerEC2Role` ARN, che hai	AWS DevOps

Operazione	Description	Competenze richieste
	copiato quando hai creato lo stack. `Prowler-Resources` • `ProwlerExecRoleName` — Mantieni il valore predefinito, a **`ProwlerExecRole`** meno che non sia stato usato un altro nome durante la distribuzione del file Prowler-Resources.yaml. 4. In Permissions (Autorizzazioni) scegliere Service-managed permissions (Autorizzazioni gestite dal servizio). 5. Nella pagina Imposta opzioni di distribuzione, in Obiettivi di distribuzione, scegli Distribuisci nell'organizzazione e accetta tutte le impostazioni predefinite. Nota: se desideri che gli stack vengano distribuiti contemporaneamente su tutti gli account membri, imposta Numero massimo di account simultanei e Tolleranza agli errori su un valore elevato, ad esempio. `100` 6. In Regioni di distribuzione, scegli Regione AWS dove	

Operazione	Description	Competenze richieste
	viene distribuita l' EC2 istanza di Prowler. Poiché le risorse IAM sono globali e non regionali, questo implementa il ruolo IAM in tutte le regioni attive. 7. Nella pagina di revisione , seleziona Riconosco che AWS CloudFormation potrebbe creare risorse IAM con nomi personalizzati, quindi scegli Crea StackSet. 8. Monitora la scheda Stack Instances (per lo stato dei singoli account) e la scheda Operations (per lo stato generale) per determinare quando la distribuzione è completa.	

Operazione	Description	Competenze richieste
Fornisci il ruolo IAM nell'account di gestione.	Utilizzando il modello IAM-ProwlerExecRole .yaml, crei uno CloudFormation stack che distribuisce il ruolo `ProwlerExecRole` IAM nell'account di gestione dell'organizzazione. Lo stack set creato in precedenza non distribuisce il ruolo IAM nell'account di gestione. Per istruzioni, consulta Creazione di uno stack nella documentazione. CloudFormation Tieni presente quanto segue durante la distribuzione di questo modello: 1. Nella pagina Specificare il modello, scegli Il modello è pronto, quindi carica il file IAM- ProwlerExecRole .yaml. 2. Nella pagina Specificare i dettagli dello stack, nella casella Nome dello stack, immettere. `IAM-ProwlerExecRole` 3. Nella sezione Parametri, inserisci quanto segue: • `AuthorizedARN` — Inserisci l'`ProwlerEC2Role` ARN, che hai copiato quando hai creato	AWS DevOps

Operazione	Description	Competenze richieste
	lo stack. `Prowler-Resources` • `ProwlerExecRoleName` — Mantieni il valore predefinito, a **`ProwlerExecRole`** meno che non sia stato usato un altro nome durante la distribuzione del file Prowler-Resources.yaml. 4. Nella pagina Revisione, seleziona Le seguenti risorse richiedono funzionalità: [], quindi scegli Create Stack. AWS::IAM::Role	

Esegui la valutazione della sicurezza di Prowler

Operazione	Description	Competenze richieste
Esegui la scansione.	1. Accedi all'account di sicurezza dell'organizzazione. 2. Utilizzando Session Manager, connettiti all' EC2 istanza di Prowler di cui hai precedentemente fornito il provisioning. Per istruzioni, consulta Connect alla tua istanza Linux usando Session Manager. Se non riesci a connetterti, consulta la sezione Risoluzione dei problemi di questo schema.	Amministratore AWS

Operazione	Description	Competenze richieste

	3. Accedere al `usr/local/prowler` file prowler_scan.sh e aprirlo.
	4. Esaminate e modificate i parametri e le variabili regolabili in questo script in base alle esigenze del vostro ambiente. Per ulteriori informazioni sulle opzioni di personalizzazione, consultate i commenti all'inizio dello script.
	
	Ad esempio, anziché ottenere un elenco di tutti gli account dei membri dell'organizzazione dall'account di gestione, è possibile modificare lo script per specificare l' Account AWS IDs o Regioni AWS che si desidera scansionare oppure fare riferimento a un file esterno che contiene questi parametri.
	5. Salvate e chiudete il file prowler_scan.sh.
	6. Esegui i comandi seguenti: Questo esegue lo script prowler_scan.sh.
	
	`sudo -i`
	`screen`	

Operazione	Description	Competenze richieste

	`cd /usr/local/prowler`
	`./prowler_scan.sh`
	
	Tenere presente quanto segue:
	• Il comando screen consente allo script di continuare l'esecuzione nel caso in cui la connessione scada o si perda l'accesso alla console.
	• Dopo l'avvio della scansione, puoi forzare il distacco dello schermo premendo Ctrl+A D. La schermata si stacca ed è possibile chiudere la connessione dell'istanza e consentire alla valutazione di procedere.
	• Per riprendere una sessione separata, connettiti all'istanza, entra `sudo -i` e poi accedi. `screen -r`
	• Per monitorare lo stato di avanzamento delle valutazioni dei singoli account, puoi accedere alla `usr/local/prowler` directory e inserire il comando. `tail`	

Operazione	Description	Competenze richieste
	`-f output/stdout-<account-id>` 7. Attendi che Prowler completi le scansioni di tutti gli account. Lo script valuta più account contempor aneamente. Quando la valutazione è completa in tutti gli account, riceverai una notifica se hai specifica to un indirizzo email quando hai distribuito il file Prowler-Resources.yaml.	

Operazione	Description	Competenze richieste
Recupera i risultati di Prowler.	1. Scarica il `prowler-output-<assessDate>.zip` file dal bucket. `prowler-output-<accountID>-<region>` Per istruzioni, consulta Download di un oggetto nella documentazione di Amazon S3. 2. Elimina tutti gli oggetti nel bucket, incluso il file scaricato. Si tratta di una procedura consigliata per l'ottimizzazione dei costi e per assicurarsi di poter eliminare lo `Prowler-Resources` CloudFormation stack in qualsiasi momento. Per istruzioni, consulta Eliminazione di oggetti nella documentazione di Amazon S3.	Informazioni generali su AWS
Interrompi l' EC2 istanza.	Per impedire la fatturazione mentre l'istanza è inattiva, interrompi l' EC2 istanza che esegue Prowler. Per istruzioni, consulta Stop and start your istances nella EC2 documentazione di Amazon.	AWS DevOps

Crea un rapporto sui risultati

Operazione	Description	Competenze richieste
Importa i risultati.	1. In Excel, apri il prowler-report-templatefile.xlsx, quindi scegli il foglio di lavoro Prowler CSV. 2. Eliminate tutti i dati di esempio, inclusa la riga di intestazione. Se vi viene chiesto se eliminare la query associata ai dati da rimuovere, scegliete No. L'eliminazione della query può influire sulla funzional ità delle tabelle pivot nel modello Excel. 3. Estrai il contenuto del file zip scaricato dal bucket S3. 4. In Excel, apri il file.txt. prowler-fullorgresults-accessdeniedfiltered Si consiglia di utilizzar e questo file perché gli errori più comuni e irrisolvi bili sono già stati rimossi, ad esempio gli `Access Denied` errori relativi ai tentativi di scansione delle risorse. AWS Control Tower Se desideri che i risultati non siano filtrati, apri invece il file prowler-fullorgres ults.txt. 5. Seleziona la colonna A.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	6. Se usi Windows, digita Ctrl+C, oppure se usi macOS, inserisci Cmd+C. Questo copia tutti i dati negli appunti. 7. Nel modello di report Excel, nel foglio di lavoro Prowler CSV, seleziona la cella A1. 8. Se usi Windows, digita Ctrl +V, oppure se usi macOS, inserisci Cmd+V. In questo modo i risultati vengono incollati nel rapporto. 9. Conferma che tutte le celle contenenti i dati incollati siano selezionate. In caso contrario, seleziona la colonna A. 10. Nella scheda Dati, scegli Testo in colonne. 11. Nella procedura guidata, procedi come segue: • Per il passaggio 1, scegli Delimitato. • Per il passaggio 2, per Delimitatori, scegli Punto e virgola. Nel riquadro di anteprima dei dati, conferma che i dati siano separati in colonne. • Per il passaggio 3, scegli Fine.	

Operazione	Description	Competenze richieste
	12. Verifica che i dati di testo siano delimitati su più colonne. 13. Salva il report Excel con un nuovo nome. 14. Cerca ed elimina eventuali `Access Denied` errori nei risultati. Per istruzioni su come rimuoverli a livello di codice, consulta Rimozione degli errori a livello di codice nella sezione Informazioni aggiuntive.	

Operazione	Description	Competenze richieste
Finalizza il rapporto.	1. Scegli il foglio di lavoro Findings, quindi seleziona la cella A17. Questa cella è l'intestazione della tabella pivot. 2. Nella barra multifunzione, in PivotTable Strumenti, scegli Analizza, quindi in Aggiorna, scegli Aggiorna tutto. Ciò aggiorna le tabelle pivot con il nuovo set di dati. 3. Per impostazione predefini ta, Excel non visualizza Account AWS correttam ente i numeri. Per corregger e la formattazione dei numeri, procedi come segue: • Nel foglio di lavoro Findings, apri il menu contestuale (fai clic con il pulsante destro del mouse) per la colonna A, quindi scegli Formato celle. • Scegli Numero e, in Posizioni decimali, inserisci. `0` • Scegli OK. Nota: se un Account AWS numero inizia con uno o più zeri, Excel rimuove	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	automaticamente gli zeri. Se nel rapporto viene visualizzato un numero di account composto da meno di 12 cifre, le cifre mancanti sono zeri all'inizio del numero. 4. (Facoltativo) Puoi comprimere i campi per facilitare la lettura dei risultati. Esegui questa operazione: • Nel foglio di lavoro Findings, se si sposta il cursore sulla riga tra le righe 18 e 19 (lo spazio tra l'intestazione critica e il primo risultato), l'icona del cursore si trasforma in una piccola freccia rivolta verso il basso. • Fai clic per selezionare tutti i campi di ricerca. • Apri il menu contestua le (fai clic con il pulsante destro del mouse), trova Espandi/Comprimi, quindi scegli Comprimi. 5. Per i dettagli sulla valutazio ne, consulta i fogli di lavoro Findings, Severity e Pass Fail.	

(Facoltativo) Aggiorna Prowler o le risorse nel repository di codice

Operazione	Description	Competenze richieste

Aggiorna Prowler.	Se desideri aggiornare Prowler alla versione più recente, procedi come segue:
	1. Connect all' EC2 istanza di Prowler utilizzando Session Manager. Per istruzioni, consulta Connect alla tua istanza Linux usando Session Manager.
	2. Inserire il seguente comando.
	`sudo -i`
	`pip3 install --upgrade prowler`
		Informazioni generali su AWS

Aggiorna lo script prowler_scan.sh.	Se desideri aggiornare lo script prowler_scan.sh all'ultima versione del repository, procedi come segue: 1. Connect all' EC2 istanza di Prowler utilizzando Session Manager. Per istruzioni, consulta Connect alla tua istanza Linux usando Session Manager. 2. Inserire il seguente comando. `sudo -i`	Informazioni generali su AWS

Operazione	Description	Competenze richieste

	3. Vai alla directory degli script di Prowler.
	`cd /usr/local/prowler`
	4. Immettete il seguente comando per nasconder e lo script locale in modo da poter unire le modifiche personalizzate nella versione più recente.
	`git stash`
	5. Immettete il seguente comando per ottenere la versione più recente dello script.
	`git pull`
	6. Immettere il comando seguente per unire lo script personalizzato alla versione più recente dello script.
	`git stash pop`
	Note
	Potresti ricevere avvisi relativi a qualsiasi file generato localment e che non si trova nel GitHub repositor	

Operazione	Description	Competenze richieste
	y, ad esempio la ricerca di report. È possibile ignorarli purché il file prowler_scan.sh mostri che le modifiche memorizzate localmente vengono nuovamente incorporate.	

(Facoltativo) Pulizia

Operazione	Description	Competenze richieste
Eliminare tutte le risorse distribuite.	È possibile lasciare le risorse distribuite negli account. Se si spegne l' EC2 istanza quando non è in uso e si mantiene il bucket S3 vuoto, si riducono i costi di manutenzione delle risorse per le scansioni future. Se desideri eseguire il deprovisioning di tutte le risorse, procedi come segue: 1. Elimina lo `IAM-ProwlerExecRole` stack fornito nell'account di gestione. Per istruzioni, consulta Eliminazione di uno stack nella documentazione. CloudFormation	AWS DevOps

Operazione	Description	Competenze richieste
	2. Elimina il set di `IAM-ProwlerExecRole` stack fornito nell'account di gestione dell'organizzazione o nell'account amministratore delegato. Per istruzioni, consulta Eliminare un set di stack nella documentazione. CloudFormation 3. Elimina tutti gli oggetti nel bucket `prowler-output` S3. Per istruzioni, consulta Eliminazione di oggetti nella documentazione di Amazon S3. 4. Elimina lo `Prowler-Resources` stack fornito nell'account di sicurezza. Per istruzioni, consulta Eliminazione di uno stack nella documentazione. CloudFormation	

risoluzione dei problemi

Problema	Soluzione
Impossibile connettersi all' EC2 istanza utilizzando Session Manager.	L'agente SSM deve essere in grado di comunicare con l'endpoint Systems Manager. Esegui questa operazione: 1. Verifica che la sottorete in cui è distribuita l' EC2 istanza abbia accesso a Internet.

Problema	Soluzione
	2. Riavviare l'istanza. EC2
Quando si distribuisce lo stack set, la CloudFormation console richiede di farlo. `Enable trusted access with AWS Organizations to use service-managed permissions`	Ciò indica che l'accesso affidabile non è stato abilitato tra e. AWS Organizations CloudFormation È necessario un accesso affidabile per distribuire il set di stack gestito dai servizi. Scegli il pulsante per abilitare l'accesso affidabile. Per ulteriori informazioni, consulta Abilitare l'accesso affidabile nella CloudFormation documentazione.

Risorse correlate

AWS documentazione

- Implementazione dei controlli di sicurezza su AWS (AWS Prescriptive Guidance)

Altre risorse

- Vagabondo () GitHub

Informazioni aggiuntive

Rimozione programmatica degli errori

Se i risultati contengono `Access Denied` errori, è necessario rimuoverli dai risultati. Questi errori sono in genere dovuti a permessi di influenza esterni che impediscono a Prowler di valutare una particolare risorsa. Ad esempio, alcuni controlli falliscono quando si esaminano i bucket S3 forniti tramite. AWS Control TowerÈ possibile estrarre questi risultati a livello di codice e salvare i risultati filtrati come nuovo file.

I comandi seguenti rimuovono le righe che contengono una singola stringa di testo (un pattern) e quindi inviano i risultati in un nuovo file.

- Per Linux o macOS (Grep)

```
grep -v -i "Access Denied getting bucket" myoutput.csv > myoutput_modified.csv
```

- Per Windows () PowerShell

```
Select-String -Path myoutput.csv -Pattern 'Access Denied getting bucket' -NotMatch >
 myoutput_modified.csv
```

I comandi seguenti rimuovono le righe che corrispondono a più di una stringa di testo e quindi restituiscono i risultati in un nuovo file.

- Per Linux o macOS (utilizza una pipe con escape tra le stringhe)

```
grep -v -i 'Access Denied getting bucket\|Access Denied Trying to Get' myoutput.csv >
 myoutput_modified.csv
```

- Per Windows (utilizza una virgola tra le stringhe)

```
Select-String -Path myoutput.csv -Pattern 'Access Denied getting bucket', 'Access
 Denied Trying to Get' -NotMatch > myoutput_modified.csv
```

Esempi di report

L'immagine seguente è un esempio del foglio di lavoro Findings contenuto nel report dei risultati consolidati di Prowler.

L'immagine seguente è un esempio del foglio di lavoro Pass Fail contenuto nel report dei risultati consolidati di Prowler. (Per impostazione predefinita, i risultati del pass sono esclusi dall'output.)

L'immagine seguente è un esempio del foglio di lavoro Severity contenuto nel report dei risultati consolidati di Prowler.

Implementa e gestisci AWS Control Tower i controlli utilizzando AWS CDK e CloudFormation

Iker Reina Fuente e Ivan Girardi, Amazon Web Services

Riepilogo

Questo modello descrive come utilizzare, implementare AWS CloudFormation e AWS Cloud Development Kit (AWS CDK) amministrare AWS Control Tower controlli preventivi, investigativi e proattivi sotto forma di infrastruttura come codice (IaC). Un controllo (noto anche come guardrail) è una regola di alto livello che fornisce una governance continua per l'intero ambiente. AWS Control Tower Ad esempio, è possibile utilizzare i controlli per richiedere la registrazione Account AWS e quindi configurare notifiche automatiche se si verificano eventi specifici relativi alla sicurezza.

AWS Control Tower ti aiuta a implementare controlli preventivi, investigativi e proattivi che regolano le tue AWS risorse e monitorano la conformità su più livelli. Account AWS Ogni controllo applica una singola regola. In questo modello, si utilizza un modello IaC fornito per specificare quali controlli si desidera implementare nel proprio ambiente.

AWS Control Tower i controlli si applicano a un'intera unità organizzativa (OU) e il controllo influisce su tutte le unità organizzative Account AWS all'interno dell'unità organizzativa. Pertanto, quando gli utenti eseguono un'azione in qualsiasi account nella tua landing zone, l'azione è soggetta ai controlli che regolano l'unità organizzativa.

L'implementazione AWS Control Tower dei controlli aiuta a stabilire una solida base di sicurezza per la tua AWS landing zone. Utilizzando questo schema per distribuire i controlli come IAc tramite CloudFormation e AWS CDK, è possibile standardizzare i controlli nella landing zone e distribuirli e gestirli in modo più efficiente. Questa soluzione utilizza cdk_nag per scansionare l'applicazione durante la distribuzione. AWS CDK Questo strumento verifica la conformità dell'applicazione alle migliori pratiche. AWS

Per implementare AWS Control Tower i controlli come IAc, puoi anche usare HashiCorp Terraform anziché. AWS CDK Per ulteriori informazioni, consulta Distribuire e gestire i AWS Control Tower controlli utilizzando Terraform.

Destinatari

Questo modello è consigliato agli utenti che hanno esperienza con AWS Control Tower CloudFormation, AWS CDK, e AWS Organizations.

Prerequisiti e limitazioni

Prerequisiti

- Active Account AWS è gestita come organizzazione all'interno AWS Organizations e come AWS Control Tower landing zone. Per istruzioni, consulta la sezione Guida introduttiva nella AWS Control Tower documentazione.
- AWS Command Line Interface (AWS CLI), installato e configurato.
- Node package manager (npm), installato e configurato per. AWS CDK
- Prerequisiti per. AWS CDK
- Autorizzazioni per assumere un ruolo esistente AWS Identity and Access Management (IAM) in un account di distribuzione.
- Autorizzazioni per assumere un ruolo IAM nell'account di gestione dell'organizzazione che può essere utilizzato per il bootstrap. AWS CDK Il ruolo deve disporre delle autorizzazioni per modificare e distribuire le risorse. CloudFormation Per ulteriori informazioni, consulta Bootstrapping nella documentazione. AWS CDK
- Autorizzazioni per creare ruoli e politiche IAM nell'account di gestione dell'organizzazione. Per ulteriori informazioni, consulta Autorizzazioni necessarie per accedere alle risorse IAM nella documentazione IAM.

Limitazioni

- Questo modello fornisce istruzioni per l'implementazione di questa soluzione Account AWS, da un account di distribuzione all'account di gestione dell'organizzazione. A scopo di test, è possibile distribuire questa soluzione direttamente nell'account di gestione, ma le istruzioni per questa configurazione non vengono fornite in modo esplicito.
- Per AWS Control Tower i controlli, questo modello richiede l'uso di identificatori globali nel seguente formato:

```
arn:<PARTITION>:controlcatalog:::control/<CONTROL_CATALOG_OPAQUE_ID>
```

  Le versioni precedenti di questo modello utilizzavano identificatori regionali che non sono più supportati. Ti consigliamo di migrare dagli identificatori regionali agli identificatori globali. Gli identificatori globali consentono di gestire i controlli e ampliare il numero di controlli utilizzabili.

> **Note**
>
> Nella maggior parte dei casi, il valore per `<PARTITION>` è`aws`.

Versioni del prodotto

- AWS Control Tower versione 3.2 o successiva
- Python versione 3.9 o successiva
- npm versione 8.9.0 o successiva

Architecture

Questa sezione fornisce una panoramica di alto livello di questa soluzione e dell'architettura stabilita dal codice di esempio. Il diagramma seguente mostra i controlli distribuiti tra i vari account dell'unità organizzativa.

AWS Control Tower i controlli sono classificati in base al loro comportamento e alle loro linee guida.

Esistono tre tipi principali di comportamenti di controllo:

1. I controlli preventivi sono progettati per impedire il verificarsi di azioni. Questi vengono implementati con politiche di controllo dei servizi (SCPs) o politiche di controllo delle risorse (RCPs) in AWS Organizations. Lo stato di un controllo preventivo è applicato o non abilitato. I controlli preventivi sono supportati in tutti Regioni AWS.
2. I controlli Detective sono progettati per rilevare eventi specifici quando si verificano e registrare l'azione AWS CloudTrail. Questi sono implementati con AWS Config regole. Lo status di un controllo investigativo è chiaro, in violazione o non abilitato. I controlli Detective si applicano solo a quelli Regioni AWS supportati da AWS Control Tower.
3. I controlli proattivi analizzano le risorse da cui verrebbero fornite AWS CloudFormation e verificano se sono conformi alle politiche e agli obiettivi aziendali. Le risorse non conformi non verranno fornite. Questi sono implementati con ganci.AWS CloudFormation Lo stato di un controllo proattivo è PASS, FAIL o SKIP.

Le linee guida al controllo si riferiscono alla pratica consigliata su come applicare ciascun controllo al proprio. OUs AWS Control Tower fornisce tre categorie di linee guida: obbligatorie, fortemente raccomandate e facoltative. La guida di un controllo è indipendente dal suo comportamento. Per ulteriori informazioni, consulta Comportamento e guida al controllo.

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che consente di definire e fornire Cloud AWS l'infrastruttura nel codice. Il AWS CDK Toolkit è lo strumento principale per interagire con la tua AWS CDK app.
- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita tra e. Account AWS Regioni AWS
- AWS Configfornisce una visione dettagliata delle risorse presenti Account AWS e di come sono configurate. Ti aiuta a identificare in che modo le risorse sono correlate tra loro e in che modo le loro configurazioni sono cambiate nel tempo.
- AWS Control Towerti aiuta a configurare e gestire un ambiente con AWS più account, seguendo le migliori pratiche prescrittive.
- AWS Organizationsè un servizio di gestione degli account che consente di consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente.

Altri strumenti

- cdk_nag è uno strumento open source che utilizza una combinazione di pacchetti di regole per verificare la conformità delle AWS CDK applicazioni alle migliori pratiche.
- npm è un registro software che viene eseguito in un ambiente Node.js e viene utilizzato per condividere o prendere in prestito pacchetti e gestire la distribuzione di pacchetti privati.
- Python è un linguaggio di programmazione per computer generico.

Archivio di codice

Il codice per questo modello è disponibile nel GitHub repository Deploy AWS Control Tower controls using AWS CDK. Si utilizza il file cdk.json per interagire con l' AWS CDK app e si utilizza il file package.json per installare i pacchetti npm.

Best practice

- Segui il principio del privilegio minimo (documentazione IAM). La policy IAM e la policy di fiducia di esempio fornite in questo modello includono le autorizzazioni minime richieste e gli AWS CDK stack creati nell'account di gestione sono limitati da queste autorizzazioni.
- Segui le migliori pratiche per AWS Control Tower gli amministratori (documentazione).AWS Control Tower
- Segui le migliori pratiche per lo sviluppo e l'implementazione dell'infrastruttura cloud con AWS CDK (AWS CDK documentazione).
- Quando avvii il AWS CDK, personalizza il modello di bootstrap per definire le politiche e gli account affidabili che dovrebbero avere la capacità di leggere e scrivere su qualsiasi risorsa dell'account di gestione. Per ulteriori informazioni, vedere Personalizzazione del bootstrap.
- Utilizzate strumenti di analisi del codice, come cfn_nag, per scansionare i modelli generati. CloudFormation Lo strumento cfn-nag cerca modelli nei CloudFormation modelli che potrebbero indicare che l'infrastruttura non è sicura. Puoi anche usare cdk-nag per controllare i tuoi CloudFormation modelli usando il modulo cloudformation-include.

Epiche

Preparati ad attivare i controlli

Operazione	Description	Competenze richieste
Crea il ruolo IAM nell'account di gestione.	1. Crea una policy IAM nell'account di gestione con le autorizzazioni definite nella policy IAM nella sezione Informazioni aggiuntive. Per istruzioni, consulta Creazione delle politiche IAM nella documentazione IAM. Prendi nota dell'Amazon Resource Name (ARN) della policy. Di seguito è	DevOps ingegnere, General AWS

Operazione	Description	Competenze richieste

	riportato un esempio di ARN.
	
	`arn:aws:iam::<MANAGEMENT-ACCOUNT-ID>:policy/<POLICY-NAME>`
	
	2. Crea un ruolo IAM nell'account di gestione, allega la politica di autorizzazione IAM creata nel passaggio precedente e allega la politica di fiducia personalizzata nella politica di fiducia nella sezione Informazioni aggiuntive. Per istruzioni, consulta Creazione di un ruolo utilizzando policy di fiducia personalizzate nella documentazione IAM. Di seguito è riportato un esempio di ARN per il nuovo ruolo.
	
	`arn:aws:iam::<MANAGEMENT-ACCOUNT-ID>:role/<ROLE-NAME>`	

Operazione	Description	Competenze richieste
Bootstrap. AWS CDK	1. Nell'account di gestione, assumi un ruolo con le autorizzazioni per il bootstrap. AWS CDK 2. Immettete il seguente comando, sostituendo il seguente: • `<MANAGEMENT-ACCOUNT-ID>` è l'ID dell'account di gestione dell'organizzazione. • `<AWS-CONTROL-TOWER-REGION>` è il Regione AWS luogo in cui AWS Control Tower viene distribuito. Per un elenco completo dei codici regionali, consulta Endpoint regionali in AWS General Reference. • `<DEPLOYMENT-ACCOUNT-ID>` è l'ID dell'account di distribuzione. • `<DEPLOYMENT-ROLE-NAME>` è il nome del ruolo IAM che stai utilizzando nell'account di distribuzione. • `<POLICY-NAME>` è il nome della policy che hai creato nell'account di gestione.	DevOps ingegnere, AWS generale, Python

Operazione	Description	Competenze richieste
	`$ npx cdk bootstrap aws://<MANAGEMENT-ACCOUNT-ID>/<AWS-CONTROL-TOWER-REGION> \ --trust arn:aws:iam::<DEPLOYMENT-ACCOUNT-ID>:role/<DEPLOYMENT-ROLE-NAME> \ --cloudformation-execution-policies arn:aws:iam::<MANAGEMENT-ACCOUNT-ID>:policy/<POLICY-NAME>`	
Clonare il repository.	In una shell bash, inserisci il seguente comando. Questo clona i AWS Control Tower controlli Deploy utilizzando AWS CDK il repository from. GitHub `git clone https://github.com/aws-samples/aws-control-tower-controls-cdk.git`	DevOps ingegnere, General AWS

Operazione	Description	Competenze richieste
Modifica il file AWS CDK di configurazione.	1. Nel repository clonato, apri il file constants.py. 2. Nel `ACCOUNT_ID` parametro, inserisci l'ID del tuo account di gestione. 3. Nel `<AWS-CONTROL-TOWER-REGION>` parametro, inserisci il Regione AWS luogo in cui AWS Control Tower viene distribuito. 4. Nel `ROLE_ARN` parametro , inserisci l'ARN del ruolo che hai creato nell'account di gestione. 5. Apri Tutti gli identificatori globali nella AWS Control Tower documentazione. 6. Nell'elenco in formato JSON, individua il controllo che desideri implement are, quindi copia il relativo identificatore globale (noto anche come valore {CONTROL_CATALOG_OPAQUE_ID}). Ad esempio, l'identificatore globale per il controllo AWS-GR_AUDIT_BUCKET_ENCRYPTION_ENABLED è. `k4izcjxhukijhajp6ks5mjxk`	DevOps ingegnere, General AWS

Operazione	Description	Competenze richieste
	7. Nella sezione, nel parametro, inserisci l'identificatore globale che hai copiato. `GUARDRAILS_CONFIGURATION` `Enable-Control` Immettete l'identificatore tra virgolette doppie e separate gli identificatori multipli con virgole. 8. Nella `GUARDRAILS_CONFIGURATION` sezione, nel `OrganizationalUnitIds` parametro, inserisci l'ID dell'unità organizzativa in cui desideri abilitare il controllo, ad esempio. `ou-1111-11111111` Inserisci l'ID tra virgolette doppie e separa i multipli IDs con virgole. Per ulteriori informazioni su come recuperare l'unità organizzativa IDs, vedere Visualizzazione dei dettagli di un'unità organizzativa. 9. Salvare e chiudere il file constants.py. Per un esempio di file constants.py aggiornato, consultate la sezione Informazioni aggiuntive di questo modello.	

Abilita i controlli nell'account di gestione

Operazione	Description	Competenze richieste
Assumi il ruolo IAM nell'account di distribuzione.	Nell'account di distribuzione, assumi il ruolo IAM che dispone delle autorizzazioni per distribuire gli AWS CDK stack nell'account di gestione. Per ulteriori informazioni sull'assunzione di un ruolo IAM in AWS CLI, consulta Utilizzare un ruolo IAM in. AWS CLI	DevOps ingegnere, General AWS

Attivare l'ambiente.	Se usi Linux o macOS:
	1. Inserisci il seguente comando per creare un ambiente virtuale.
	`$ python3 -m venv .venv`
	2. Dopo aver creato l'ambiente virtuale, inserisci il seguente comando per attivarlo.
	`$ source .venv/bin/activate`
	Se utilizzi Windows:
	1. Inserisci il seguente comando per attivare un ambiente virtuale.
		DevOps ingegnere, General AWS

Operazione	Description	Competenze richieste
	`% .venv\Scripts\activate.bat`	
Installa le dipendenze.	Dopo aver attivato l'ambiente virtuale, immettete il seguente comando per eseguire lo script install_deps.sh. Questo script installa le dipendenze richieste . `$ ./scripts/install_deps.sh`	DevOps ingegnere, AWS generale, Python

Implementa lo stack.	Inserisci i seguenti comandi per sintetizzare e distribuire lo stack. CloudFormation
	`$ npx cdk synth`
	`$ npx cdk deploy`
		DevOps ingegnere, AWS generale, Python

Risorse correlate

AWS documentazione

- Informazioni sui controlli (AWS Control Tower documentazione)
- Libreria di controlli (AWS Control Tower documentazione)
- AWS CDK Comandi Toolkit (AWS CDK documentazione)
- Implementa e gestisci i AWS Control Tower controlli utilizzando Terraform (Prescriptive Guidance)AWS

Altre risorse

- Python

Informazioni aggiuntive

Esempio di file constants.py

Di seguito è riportato un esempio di file constants.py aggiornato. Questo esempio abilita il controllo AWS-GR_DISALLOW_CROSS_REGION_NETWORKING (ID globale:) e il controllo AWS-GR_SUBNET_AUTO_ASSIGN_PUBLIC_IP_DISABLED (ID globale:). **`dvuaav61i5cnfazfelmvn9m6k`** `50z1ot237wl8u1lv5ufau6qqo` Per un elenco di identificatori globali, IDs consulta Tutti AWS Control Tower gli identificatori globali nella documentazione.

```
ACCOUNT_ID = 111122223333
AWS_CONTROL_TOWER_REGION = us-east-2
ROLE_ARN = "arn:aws:iam::111122223333:role/CT-Controls-Role"
GUARDRAILS_CONFIGURATION = [
    {
        "Enable-Control": {
            "dvuaav61i5cnfazfelmvn9m6k": {  # AWS-GR_DISALLOW_CROSS_REGION_NETWORKING
                "Parameters": {
                    "ExemptedPrincipalArns": ["arn:aws:iam::111122223333:role/
RoleName"]
                },
                "Tags": [{"key": "Environment", "value": "Production"}]
            },
            ...
        },
        "OrganizationalUnitIds": ["ou-1111-11111111", "ou-2222-22222222"...],
    },
    {
        "Enable-Control": {
            "50z1ot237wl8u1lv5ufau6qqo",  # AWS-
GR_SUBNET_AUTO_ASSIGN_PUBLIC_IP_DISABLED
            ...
        },
        "OrganizationalUnitIds": ["ou-2222-22222222"...],
    },
]
```

Policy IAM

La seguente policy di esempio consente le azioni minime richieste per abilitare o disabilitare AWS Control Tower i controlli durante la distribuzione AWS CDK degli stack da un account di distribuzione all'account di gestione.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "controltower:EnableControl",
                "controltower:DisableControl",
                "controltower:GetControlOperation",
                "controltower:ListEnabledControls",
                "organizations:AttachPolicy",
                "organizations:CreatePolicy",
                "organizations:DeletePolicy",
                "organizations:DescribeOrganization",
                "organizations:DescribeOrganizationalUnit",
                "organizations:DetachPolicy",
                "organizations:ListAccounts",
                "organizations:ListAWSServiceAccessForOrganization",
                "organizations:ListChildren",
                "organizations:ListOrganizationalUnitsForParent",
                "organizations:ListParents",
                "organizations:ListPoliciesForTarget",
                "organizations:ListRoots",
                "organizations:UpdatePolicy",
                "ssm:GetParameters"
            ],
            "Resource": "*"
        }
    ]
}
```

Policy di trust

La seguente policy di fiducia personalizzata consente a un ruolo IAM specifico nell'account di distribuzione di assumere il ruolo IAM nell'account di gestione. Sostituisci quanto segue:

- `<DEPLOYMENT-ACCOUNT-ID>`è l'ID dell'account di distribuzione
- `<DEPLOYMENT-ROLE-NAME>`è il nome del ruolo nell'account di distribuzione a cui è consentito assumere il ruolo nell'account di gestione

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::<DEPLOYMENT-ACCOUNT-ID>:role/<DEPLOYMENT-ROLE-
NAME>"
            },
            "Action": "sts:AssumeRole",
            "Condition": {}
        }
    ]
}
```

Implementa e gestisci AWS Control Tower i controlli utilizzando Terraform

Iker Reina Fuente e Ivan Girardi, Amazon Web Services

Riepilogo

Questo modello descrive come utilizzare AWS Control Tower i controlli, HashiCorp Terraform e l'infrastruttura come codice (IaC) per implementare e amministrare controlli di sicurezza preventivi, investigativi e proattivi. Un controllo (noto anche come guardrail) è una regola di alto livello che fornisce una governance continua per l'intero ambiente. AWS Control Tower Ad esempio, è possibile utilizzare i controlli per richiedere la registrazione Account AWS e quindi configurare notifiche automatiche se si verificano eventi specifici relativi alla sicurezza.

AWS Control Tower ti aiuta a implementare controlli preventivi, investigativi e proattivi che regolano le tue AWS risorse e monitorano la conformità su più livelli. Account AWS Ogni controllo applica una singola regola. In questo modello, si utilizza un modello IaC fornito per specificare quali controlli si desidera implementare nel proprio ambiente.

AWS Control Tower i controlli si applicano a un'intera unità organizzativa (OU) e il controllo influisce su tutte le unità organizzative Account AWS all'interno dell'unità organizzativa. Pertanto, quando gli utenti eseguono un'azione in qualsiasi account nella tua landing zone, l'azione è soggetta ai controlli che regolano l'unità organizzativa.

L'implementazione AWS Control Tower dei controlli aiuta a stabilire una solida base di sicurezza per la tua AWS landing zone. Utilizzando questo modello per implementare i controlli come IAc tramite Terraform, puoi standardizzare i controlli nella tua landing zone e distribuirli e gestirli in modo più efficiente.

Per distribuire i AWS Control Tower controlli come IAc, puoi anche utilizzare al posto di Terraform. AWS Cloud Development Kit (AWS CDK) Per ulteriori informazioni, consulta Distribuire e gestire i AWS Control Tower controlli utilizzando e. AWS CDK AWS CloudFormation

Destinatari

Questo modello è consigliato agli utenti che hanno esperienza con AWS Control Tower, Terraform e AWS Organizations.

Prerequisiti e limitazioni

Prerequisiti

- Active Account AWS è gestita come organizzazione all'interno AWS Organizations e come AWS Control Tower landing zone. Per istruzioni, consulta la sezione Guida introduttiva nella AWS Control Tower documentazione.
- AWS Command Line Interface (AWS CLI), installato e configurato.
- Un ruolo AWS Identity and Access Management (IAM) nell'account di gestione che dispone delle autorizzazioni per implementare questo pattern. Per ulteriori informazioni sulle autorizzazioni richieste e una policy di esempio, consulta Least Privilege permissions for the IAM role nella sezione Informazioni aggiuntive di questo modello.
- Autorizzazioni per assumere il ruolo IAM nell'account di gestione.
- Terraform CLI, installata (documentazione Terraform).
- Terraform AWS Provider, configurato (documentazione Terraform).
- Backend Terraform, configurato (documentazione Terraform).

Limitazioni

- Per AWS Control Tower i controlli, questo modello richiede l'uso di identificatori globali nel seguente formato:

```
arn:<PARTITION>:controlcatalog:::control/<CONTROL_CATALOG_OPAQUE_ID>
```

  Le versioni precedenti di questo modello utilizzavano identificatori regionali che non sono più supportati. Ti consigliamo di migrare dagli identificatori regionali agli identificatori globali. Gli identificatori globali consentono di gestire i controlli e ampliare il numero di controlli utilizzabili.

> **Note**
>
> Nella maggior parte dei casi, il valore per `<PARTITION>` è`aws`.

Versioni del prodotto

- AWS Control Tower versione 3.2 o successiva

- Terraform versione 1.5 o successiva
- Terraform AWS Provider versione 4.67 o successiva

Architecture

Questa sezione fornisce una panoramica di alto livello di questa soluzione e dell'architettura stabilita dal codice di esempio. Il diagramma seguente mostra i controlli distribuiti tra i vari account dell'unità organizzativa.

AWS Control Tower i controlli sono classificati in base al loro comportamento e alle loro linee guida.

Esistono tre tipi principali di comportamenti di controllo:

1. I controlli preventivi sono progettati per impedire il verificarsi di azioni. Questi vengono implementati con politiche di controllo dei servizi (SCPs) o politiche di controllo delle risorse (RCPs) in AWS Organizations. Lo stato di un controllo preventivo è applicato o non abilitato. I controlli preventivi sono supportati in tutti Regioni AWS.
2. I controlli Detective sono progettati per rilevare eventi specifici quando si verificano e registrare l'azione AWS CloudTrail. Questi sono implementati con AWS Config regole. Lo status di un controllo investigativo è chiaro, in violazione o non abilitato. I controlli Detective si applicano solo a quelli Regioni AWS supportati da AWS Control Tower.
3. I controlli proattivi analizzano le risorse da cui verrebbero fornite AWS CloudFormation e verificano se sono conformi alle politiche e agli obiettivi aziendali. Le risorse non conformi non verranno fornite. Questi sono implementati con ganci.AWS CloudFormation Lo stato di un controllo proattivo è PASS, FAIL o SKIP.

La guida al controllo è la pratica consigliata per applicare ogni controllo al tuo. OUs AWS Control Tower fornisce tre categorie di linee guida: obbligatorie, fortemente consigliate e facoltative. La guida di un controllo è indipendente dal suo comportamento. Per ulteriori informazioni, consulta Comportamento e guida al controllo.

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita in tutte Account AWS le regioni.

- AWS Configfornisce una visione dettagliata delle risorse disponibili Account AWS e di come sono configurate. Ti aiuta a identificare in che modo le risorse sono correlate tra loro e in che modo le loro configurazioni sono cambiate nel tempo.
- AWS Control Towerti aiuta a configurare e gestire un ambiente con AWS più account, seguendo le migliori pratiche prescrittive.
- AWS Organizationsè un servizio di gestione degli account che consente di consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente.

Altri strumenti

- HashiCorp Terraform è uno strumento Infrastructure as Code (IaC) che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud.

Deposito di codici

Il codice per questo modello è disponibile nel GitHub Deploy and manage AWS Control Tower controls utilizzando il repository Terraform.

Best practice

- Il ruolo IAM utilizzato per implementare questa soluzione deve rispettare il principio del privilegio minimo (documentazione IAM).
- Segui le migliori pratiche per gli amministratori (documentazione). AWS Control Tower AWS Control Tower

Epiche

Abilita i controlli nell'account di gestione

Operazione	Description	Competenze richieste
Clonare il repository.	In una shell bash, inserisci il seguente comando. Questo clona i AWS Control Tower controlli Deploy and manage	DevOps ingegnere

Operazione	Description	Competenze richieste
	utilizzando il repository Terraform da. GitHub `git clone https://github.com/aws-samples/aws-control-tower-controls-terraform.git`	
Modifica il file di configurazione del backend Terraform.	1. Nel repository clonato, apri il file backend.tf. 2. Modifica il file per impostare la configurazione del backend Terraform. La configurazione definita in questo file dipende dal tuo ambiente. Per ulteriori informazioni, consulta Configurazione del backend (documentazione Terraform). 3. Salva e chiudi il file backend.tf.	DevOps ingegnere, Terraform

Operazione	Description	Competenze richieste
Modifica il file di configura zione del provider Terraform.	1. Nel repository clonato, apri il file provider.tf. 2. Modifica il file per impostare la configurazione del provider Terraform. Per ulteriori informazioni, vedere Configurazione del provider (document azione Terraform). Imposta Regione AWS come regione in cui è disponibile l' AWS Control Tower API. 3. Salva e chiudi il file provider.tf.	DevOps ingegnere, Terraform

Operazione	Description	Competenze richieste
Modifica il file di configurazione.	1. Nel repository clonato, apri il file variables.tfvars. 2. Apri Tutti gli identificatori globali AWS Control Tower nella documentazione. 3. Nell'elenco in formato JSON, individua il controllo che desideri implementare, quindi copia il relativo identificatore globale (noto anche come valore {CONTROL_CATALOG_OPAQUE_ID}). Ad esempio, l'identificatore globale per il controllo AWS-GR_AUDIT_BUCKET_ENCRYPTION_ENABLED è. `k4izcjxhukijhajp6ks5mjxk` 4. Nella sezione, nel parametro, inserisci l'identificatore globale che hai copiato. `controls control_names` 5. Nella `controls` sezione, nel `organizational_unit_ids` parametro, inserisci l'ID dell'unità organizzativa in cui desideri abilitare il controllo, ad esempio. `ou-1111-11111111` Inserisci l'ID tra virgolette doppie e separa i multipli	DevOps ingegnere, General AWS, Terraform

Operazione	Description	Competenze richieste
	IDs con virgole. Per ulteriori informazioni su come recuperare l'unità organizzativa IDs, vedere Visualizzazione dei dettagli di un'unità organizzativa. 6. Salvare e chiudere il file variables.tfvars. Per un esempio di file variables.tfvars aggiornato, consultate la sezione Informazioni aggiuntive di questo modello.	
Assumi il ruolo IAM nell'account di gestione.	Nell'account di gestione, assumi il ruolo IAM che dispone delle autorizzazioni per distribuire il file di configurazione Terraform. Per ulteriori informazioni sulle autorizzazioni richieste e una politica di esempio, consulta le autorizzazioni con privilegi minimi per il ruolo IAM nella sezione Informazioni aggiuntive. Per ulteriori informazioni sull'assunzione di un ruolo IAM in AWS CLI, consulta Utilizzare un ruolo IAM in. AWS CLI	DevOps ingegnere, General AWS

Operazione	Description	Competenze richieste

Implementa il file di configurazione.	1. Immettere il seguente comando per inizializzare Terraform.
	`$ terraform init -upgrade`
	2. Immettere il seguente comando per visualizzare in anteprima le modifiche rispetto allo stato corrente.
	`$ terraform plan -var-file="variables.tfvars"`
	3. Rivedi le modifiche alla configurazione nel piano Terraform e conferma che desideri implementare queste modifiche nell'organizzazione.
	4. Immettere il seguente comando per distribuire le risorse.
	`$ terraform apply -var-file="variables.tfvars"`
		DevOps ingegnere, General AWS, Terraform

(Facoltativo) Disattiva i controlli nell'account di AWS Control Tower gestione

Operazione	Description	Competenze richieste
Esegui il comando `destroy`.	Immettete il seguente comando per rimuovere le risorse distribuite secondo questo schema. `$ terraform destroy -var-file="variables.tfvars"`	DevOps ingegnere, General AWS, Terraform

risoluzione dei problemi

Problema	Soluzione
Errore `Error: creating ControlTower Control ValidationException: Guardrail <control ID> is already enabled on organizational unit <OU ID>`	Il controllo che stai cercando di abilitare è già abilitato nell'unità organizzativa di destinazione. Questo errore può verificarsi se un utente ha abilitato manualmente il controllo tramite Console di gestione AWS, through AWS Control Tower o through AWS Organizations. Per distribuire il file di configurazione Terraform, puoi utilizzare una delle seguenti opzioni. Opzione 1: aggiorna il file dello stato corrente di Terraform È possibile importare la risorsa nel file dello stato corrente di Terraform. Quando riesegui il `apply` comando, Terraform salterà questa risorsa. Effettua quanto segue per importare la risorsa nello stato corrente di Terraform: 1. Nell'account di AWS Control Tower gestione, inserisci il seguente comando per recuperar

Problema	Soluzione

	e un elenco di Amazon Resource Names (ARNs) per OUs, where `<root-ID>` è la radice dell'organizzazione. Per ulteriori informazioni sul recupero di questo ID, consulta Visualizzazione dei dettagli della radice.
	
	`aws organizations list-organizational-units-for-parent --parent-id <root-ID>`
	
	2. Per ogni unità organizzativa restituita nel passaggio precedente, immettere il comando seguente, dove <OU-ARN> è l'ARN dell'unità organizzativa.
	
	`aws controltower list-enabled-controls --target-identifier <OU-ARN>`
	
	3. Copia ARNs ed esegui l'importazione di Terraform nel modulo richiesto in modo che sia incluso nello stato Terraform. Per istruzioni, consulta Import (documentazione Terraform).
	4. Ripeti i passaggi in Implementa la configurazione nella sezione Epics.
	
	Opzione 2: disabilita il controllo
	
	Se lavori in un ambiente non di produzione, puoi disabilitare il controllo nella console. Riattivalo ripetendo i passaggi descritti in Deploy the configuration nella sezione Epics. Questo approccio non è consigliato per gli ambienti di produzione perché c'è un periodo di tempo in cui il controllo sarà disabilitato.

Problema	Soluzione
	Se si desidera utilizzare questa opzione in un ambiente di produzione, è possibile implement are controlli temporanei, come l'applicazione temporanea di un SCP in AWS Organizations.

Risorse correlate

AWS documentazione

- Informazioni sui controlli (AWS Control Tower documentazione)
- Libreria di controlli (AWS Control Tower documentazione)
- Implementa e gestisci AWS Control Tower i controlli utilizzando AWS CDK and AWS CloudFormation (AWS Prescriptive Guidance)

Altre risorse

- Terraform
- Documentazione CLI Terraform

Informazioni aggiuntive

Esempio di file variables.tfvars

Di seguito è riportato un esempio di file variables.tfvars aggiornato. Questo esempio abilita il controllo AWS-GR_ENCRYPTED_VOLUMES (ID globale:) e il controllo AWS-GR_SUBNET_AUTO_ASSIGN_PUBLIC_IP_DISABLED (ID globale:**`503uicglhjkokaajywfpt6ros`**). `50z1ot237wl8u1lv5ufau6qqo` Per un elenco di identificatori IDs globali, consulta Tutti AWS Control Tower gli identificatori globali nella documentazione.

L'esempio seguente abilita anche controlli con parametri come CT.S3.PV.5 (global ID:**`7mo7a2h2ebsq71l8k6uzr96ou`**) e CT.SECRETSMANAGER.PV.1 (global ID:). `dvhe47fxg5o6lryqrq9g6sxg4` Per un elenco di controlli con parametri, vedete Controlli con parametri nella documentazione. AWS Control Tower

```
controls = [
    {
        control_names = [
            "503uicglhjkokaajywfpt6ros", # AWS-GR_ENCRYPTED_VOLUMES
            ...
        ],
        organizational_unit_ids = ["ou-1111-11111111", "ou-2222-22222222"...],
    },
    {
        control_names = [
            "50z1ot237wl8u1lv5ufau6qqo", # AWS-GR_SUBNET_AUTO_ASSIGN_PUBLIC_IP_DISABLED
            ...
        ],
        organizational_unit_ids = ["ou-1111-11111111"...],
    },
]

controls_with_params = [
  {
    control_names = [
      { "7mo7a2h2ebsq71l8k6uzr96ou" = { # CT.S3.PV.5
        parameters = {
          "ExemptedPrincipalArns" : ["arn:aws:iam::*:role/RoleName"],
          "ExemptedResourceArns" : [],
        }
      } },
      { "dvhe47fxg5o6lryqrq9g6sxg4" = { # CT.SECRETSMANAGER.PV.1
        parameters = {
          "ExemptedPrincipalArns" : ["arn:aws:iam::*:role/RoleName"],
        }
      } },
      ...
    ],
    organizational_unit_ids = ["ou-1111-11111111"...]
  },
  {
    control_names = [
      { "dvuaav61i5cnfazfelmvn9m6k" = { # AWS-GR_DISALLOW_CROSS_REGION_NETWORKING
        parameters = {
          "ExemptedPrincipalArns" : ["arn:aws:iam::*:role/RoleName"],
        }
      } },
      { "41ngl8m5c4eb1myoz0t707n7h" = { # AWS-GR_DISALLOW_VPC_INTERNET_ACCESS
```

```
        parameters = {
          "ExemptedPrincipalArns" : ["arn:aws:iam::*:role/RoleName"],
        }
      } },
      ...
    ],
    organizational_unit_ids = ["ou-2222-22222222"...]
  }
]
```

Autorizzazioni con privilegi minimi per il ruolo IAM

Questo modello richiede l'assunzione di un ruolo IAM nell'account di gestione. La migliore pratica consiste nell'assumere un ruolo con autorizzazioni temporanee e limitare le autorizzazioni in base al principio del privilegio minimo. La seguente politica di esempio consente le azioni minime richieste per abilitare o disabilitare i controlli. AWS Control Tower

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "controltower:EnableControl",
                "controltower:DisableControl",
                "controltower:GetControlOperation",
                "controltower:ListEnabledControls",
                "organizations:AttachPolicy",
                "organizations:CreatePolicy",
                "organizations:DeletePolicy",
                "organizations:DescribeOrganization",
                "organizations:DetachPolicy",
                "organizations:ListAccounts",
                "organizations:ListAWSServiceAccessForOrganization",
                "organizations:ListChildren",
                "organizations:ListOrganizationalUnitsForParent",
                "organizations:ListParents",
                "organizations:ListPoliciesForTarget",
                "organizations:ListRoots",
                "organizations:UpdatePolicy"
            ],
            "Resource": "*"
        }
```

```
    ]
}
```

Implementa le automazioni di sicurezza per la AWS WAF soluzione utilizzando Terraform

Dott. Rahul Sharad Gaikwad e Tamilselvan P, Amazon Web Services

Riepilogo

AWS WAF è un firewall per applicazioni Web che aiuta a proteggere le applicazioni dagli exploit comuni utilizzando regole personalizzabili, che vengono definite e implementate nelle liste di controllo degli accessi Web (). ACLs La configurazione AWS WAF delle regole può essere impegnativa, soprattutto per le organizzazioni che non dispongono di team di sicurezza dedicati. Per semplificare questo processo, Amazon Web Services (AWS) offre la AWS WAF soluzione Security Automations for, che implementa automaticamente un singolo ACL Web con una serie di AWS WAF regole che filtrano gli attacchi basati sul Web. Durante l'implementazione di Terraform, puoi specificare quali funzionalità di protezione includere. Dopo aver distribuito questa soluzione, AWS WAF ispeziona le richieste Web alle CloudFront distribuzioni Amazon esistenti o agli Application Load Balancer e blocca tutte le richieste che non corrispondono alle regole.

La AWS WAF soluzione Security Automations for può essere implementata AWS CloudFormation seguendo le istruzioni contenute nella Security Automations for Implementation Guide. AWS WAF Questo modello fornisce un'opzione di implementazione alternativa per le organizzazioni che utilizzano HashiCorp Terraform come strumento preferito di infrastruttura come codice (IaC) per fornire e gestire la propria infrastruttura cloud. Quando si implementa questa soluzione, Terraform applica automaticamente le modifiche nel cloud e implementa e configura le impostazioni e le funzionalità di protezione. AWS WAF

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS
- AWS Command Line Interface (AWS CLI) versione 2.4.25 o successiva, installata e configurata con le autorizzazioni necessarie. Per ulteriori informazioni, vedere Guida introduttiva (documentazione)AWS CLI .
- Terraform versione 1.1.9 o successiva, installata e configurata. Per ulteriori informazioni, consulta Install Terraform (documentazione Terraform).

Architecture

Architettura Target

Questo modello implementa la soluzione Security Automations for AWS WAF . Per ulteriori informazioni sull'architettura di destinazione, consulta la panoramica dell'architettura nella Security Automations for AWS WAF Implementation Guide. Per ulteriori informazioni sulle AWS Lambda automazioni in questa distribuzione, sull'Application log parser, sul log parser, sul parser degli AWS WAF elenchi IP e sul gestore di accesso, consulta i dettagli dei componenti nella Security Automations for Implementation Guide. AWS WAF

Distribuzione di Terraform

Quando corri`terraform apply`, Terraform esegue le seguenti operazioni:

1. Terraform crea ruoli AWS Identity and Access Management (IAM) e funzioni Lambda in base agli input del file testing.tfvars.
2. Terraform crea regole AWS WAF ACL e set IP in base agli input del file testing.tfvars.
3. Terraform crea i bucket Amazon Simple Storage Service (Amazon S3), le regole AWS Glue Amazon, le tabelle di database e i gruppi di lavoro EventBridge Amazon Athena in base agli input del file testing.tfvars.
4. Terraform distribuisce lo stack per fornire le risorse personalizzate. AWS CloudFormation
5. Terraform crea le risorse Amazon API Gateway in base agli input forniti dal file testing.tfvars.

Automazione e scalabilità

È possibile utilizzare questo modello per creare AWS WAF regole per più utenti Account AWS e Regioni AWS distribuire la AWS WAF soluzione Security Automations for in tutto l' Cloud AWS ambiente.

Tools (Strumenti)

Servizi AWS

- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando.
- AWS WAFè un firewall per applicazioni Web che consente di monitorare le richieste HTTP e HTTPS inoltrate alle risorse protette delle applicazioni Web.

Altri servizi

- Git è un sistema di controllo delle versioni distribuito e open source.
- HashiCorp Terraform è un'applicazione di interfaccia a riga di comando che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud.

Archivio di codice

Il codice per questo pattern è disponibile nel repository GitHub AWS WAF Automation Using Terraform.

Best practice

- Inserisci file statici in bucket Amazon S3 separati.
- Evita le variabili di codifica rigida.
- Limita l'uso di script personalizzati.
- Adotta una convenzione di denominazione.

Epiche

Configura la tua workstation locale

Operazione	Description	Competenze richieste
Installa Git.	Segui le istruzioni in Guida introduttiva (sito Web Git) per installare Git sulla tua workstation locale.	DevOps ingegnere
Clonare il repository.	Sulla tua workstation locale, inserisci il seguente comando per clonare il repository di codice: `git clone https://github.com/aws-samples/aws-waf-automa`	DevOps ingegnere

Operazione	Description	Competenze richieste
	`tion-terraform-sam ples.git`	
Aggiorna le variabili.	1. Naviga nella directory clonata inserendo il seguente comando: `cd terraform-aws-waf-automation` 2. In qualsiasi editor di testo, apri il file testing.tfvars. 3. Aggiorna i valori delle variabili nel file testing.t fvars. 4. Salva e chiudi il file.	DevOps ingegnere

Fornisci l'architettura di destinazione utilizzando Terraform

Operazione	Description	Competenze richieste
Inizializza la configurazione Terraform.	Inserisci il seguente comando per inizializzare la tua directory di lavoro che contiene i file di configurazione Terraform: `terraform init`	DevOps ingegnere
Visualizza l'anteprima del piano Terraform.	Inserire il seguente comando. Terraform valuta i file di configurazione per determina re lo stato di destinazione delle risorse dichiarate. Quindi confronta lo stato di destinazi	DevOps ingegnere

Operazione	Description	Competenze richieste
	one con lo stato attuale e crea un piano: `terraform plan -var-file ="testing.tfvars"`	
Verifica il piano.	Rivedi il piano e conferma che configuri l'architettura richiesta nel tuo obiettivo Account AWS.	DevOps ingegnere
Distribuire la soluzione.	1. Inserisci il seguente comando per applicare il piano: `terraform apply -var-file="testing .tfvars"` 2. Immettere `yes` per confermare. Terraform crea, aggiorna o distrugge l'infrastruttura per raggiunge re lo stato di destinazi one dichiarato nei file di configurazione. Per ulteriori informazioni sulla sequenza, vedere l'implementazione di Terraform nella sezione Architettura di questo modello.	DevOps ingegnere

Convalida e ripulisci

Operazione	Description	Competenze richieste
Verifica le modifiche.	1. Nella console Terraform , verifica che gli output corrispondano ai risultati previsti. 2. Accedi alla Console di gestione AWS. 3. Verifica che gli output della console Terraform siano stati distribuiti correttamente nel tuo. Account AWS	DevOps ingegnere
(Facoltativo) Pulisci l'infrastruttura.	Se desideri rimuovere tutte le risorse e le modifiche alla configurazione apportate da questa soluzione, procedi come segue: 1. Nella console Terraform , inserisci il seguente comando: `terraform destroy -var-file="testing.tfvars"` 2. Immettere `yes` per confermare.	DevOps ingegnere

risoluzione dei problemi

Problema	Soluzione
Errore `WAFV2 IPSet: WAFOptimisticLockException`	Se si riceve questo errore quando si esegue il `terraform destroy` comando, è necessario eliminare manualmente i set IP. Per istruzioni, vedere Eliminazione di un set IP (AWS WAF documentazione).

Risorse correlate

AWS riferimenti

- Guida all' AWS WAF implementazione delle automazioni di sicurezza
- Automazioni di sicurezza per AWS WAF(Libreria di AWS soluzioni)
- Automazioni di sicurezza per domande frequenti AWS WAF

Riferimenti Terraform

- Configurazione del backend Terraform
- Terraform AWS Provider - Documentazione e utilizzo
- Terraform AWS Provider (repository) GitHub

Implementa una pipeline che rilevi simultaneamente i problemi di sicurezza in più risultati di codice

Benjamin Morris, Tim Hahn, Sapeksh Madan, Dina Odum e Isaiah Schisler, Amazon Web Services

Riepilogo

La Simple Code Scanning Pipeline (SCSP) consente la creazione in due clic di una pipeline di analisi del codice che esegue in parallelo strumenti di sicurezza open source standard del settore. Ciò consente agli sviluppatori di verificare la qualità e la sicurezza del proprio codice senza dover installare strumenti o persino capire come eseguirli. Ciò consente di ridurre le vulnerabilità e le configurazioni errate nei risultati finali del codice. Riduce inoltre il tempo impiegato dall'organizzazione per l'installazione, la ricerca e la configurazione degli strumenti di sicurezza.

Prima di SCSP, la scansione del codice con questa particolare suite di strumenti richiedeva agli sviluppatori di individuare, installare e configurare manualmente gli strumenti di analisi del software. Anche se installati localmente, all-in-one gli strumenti, come Automated Security Helper (ASH), richiedono la configurazione di un contenitore Docker per funzionare. Tuttavia, con SCSP, una suite di strumenti di analisi del codice standard del settore viene eseguita automaticamente in. Cloud AWS Con questa soluzione, usi Git per inviare i risultati del codice e poi ricevi un output visivo con at-a-glance informazioni dettagliate su quali controlli di sicurezza non sono riusciti.

Prerequisiti e limitazioni

- Un attivo Account AWS
- Uno o più risultati di codice che desideri scansionare per individuare eventuali problemi di sicurezza
- AWS Command Line Interface (AWS CLI), installato e configurato
- Python versione 3.0 o successiva e versione pip 9.0.3 o successiva, installate
- Git, installato
- Installa git-remote-codecommitsulla tua workstation locale

Architecture

Stack tecnologico Target

- AWS CodeCommit deposito
- AWS CodeBuild progetto
- AWS CodePipeline oleodotto
- Bucket Amazon Simple Storage Service (Amazon S3)
- AWS CloudFormation modello

Architettura di destinazione

L'SCSP per l'analisi statica del codice è un DevOps progetto progettato per fornire feedback sulla sicurezza sul codice consegnabile.

1. In Console di gestione AWS, accedi alla destinazione Account AWS. Conferma di trovarti nel Regione AWS punto in cui desideri implementare la pipeline.
2. Usa il CloudFormation modello nel repository del codice per distribuire lo stack SCSP. Questo crea un nuovo repository e un nuovo progetto CodeCommit . CodeBuild

   > **Note**
   >
   > Come opzione di distribuzione alternativa, puoi utilizzare un CodeCommit repository esistente fornendo l'Amazon Resource Name (ARN) del repository come parametro durante la distribuzione dello stack.

3. Clona il repository sulla tua workstation locale, quindi aggiungi tutti i file alle rispettive cartelle nel repository clonato.
4. Usa Git per aggiungere, eseguire il commit e inviare i file al CodeCommit repository.
5. L'invio al CodeCommit repository avvia un processo. CodeBuild Il CodeBuild progetto utilizza gli strumenti di sicurezza per scansionare i risultati del codice.
6. Esamina l'output della pipeline. Gli strumenti di sicurezza che hanno rilevato problemi a livello di errore comporteranno il fallimento delle azioni nella pipeline. Correggi questi errori o eliminali come falsi positivi. Esamina i dettagli dell'output dello strumento nei dettagli dell'azione nel bucket S3 della pipeline CodePipeline o nel bucket S3.

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita in tutte le regioni. Account AWS
- AWS CodeBuildè un servizio di compilazione completamente gestito che consente di compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per l'implementazione.
- AWS CodeCommitè un servizio di controllo delle versioni che consente di archiviare e gestire in modo privato gli archivi Git, senza dover gestire il proprio sistema di controllo del codice sorgente.

Altri strumenti

Per un elenco completo degli strumenti utilizzati da SCSP per scansionare i risultati del codice, consultate il file readme di SCSP in. GitHub

Archivio di codice

Il codice per questo pattern è disponibile nel repository Simple Code Scanning Pipeline (SCSP) in. GitHub

Epiche

Implementa SCSP

Operazione	Description	Competenze richieste
Crea lo CloudFormation stack.	1. Accedi alla Console di gestione AWS. 2. Nella console, conferma di trovarti nella regione di destinazione in cui desideri implementare la soluzione . Per ulteriori informazi oni, consulta Scelta di una regione. 3. Scegli il seguente link. Verrà aperta la procedura guidata di creazione rapida dello stack in. CloudForm ation	AWS DevOps, amministratore AWS

Operazione	Description	Competenze richieste
	https://console.aws.amazon.com/cloudformation/casa? #/ stacks/create/review?templateURL=https://proservetools.s3.amazonaws.com/cft/scsp -pipeline-stack.template.json&stackname=SimpleCodeScanPipeline 4. Nella procedura guidata Quick create stack, rivedi le impostazioni dei parametri per lo stack e apporta le modifiche necessarie per il tuo caso d'uso. 5. Seleziona Riconosco che AWS CloudFormation potrebbe creare risorse IAM, quindi scegli Create stack. Questo crea un CodeCommit repository, una CodePipeline pipeline, diverse definizioni di CodeBuild job e un bucket S3. Le esecuzioni di compilazione e i risultati delle scansioni vengono copiati in questo bucket. Dopo che lo CloudFormation stack è stato completamente distribuito, SCSP è pronto per l'uso.	

Usa la pipeline

Operazione	Description	Competenze richieste
Esamina i risultati della scansione.	1. Nella console Amazon S3, in Buckets, scegli il bucket simplecodescanpipeline-deleteresourcespipelineresо. 2. Scegli la directory scan_results, quindi scegli la cartella con la data di scansione più recente. 3. Esamina i file di registro in questa cartella per esaminare eventuali problemi rilevati dagli strumenti di sicurezza utilizzati nella pipeline. Gli strumenti di sicurezza che hanno rilevato problemi a livello di errore comporteranno `failed` azioni in fase di implementazione. Questi devono essere corretti o eliminati se si tratta di falsi positivi. **Note** È inoltre possibile visualizzare i dettagli dell'output dello strumento (sia per le scansioni passate che per quelle non riuscite)	Sviluppatore di app, AWS DevOps

Operazione	Description	Competenze richieste
	nella CodePipel ine console, nella sezione Dettagli dell'azione.	

risoluzione dei problemi

Problema	Soluzione
HashiCorp Terraform o AWS CloudFormation i file non vengono scansionati.	Assicurati che i file Terraform (.tf) e CloudForm ation (.yml, .yaml o .json) siano inseriti nelle cartelle appropriate del repository clonato. CodeCommit
`git clone`Il comando non riesce.	Assicurati di aver installato `git-remot e-codecommit` e che la tua CLI abbia accesso alle AWS credenziali che dispongon o delle autorizzazioni per leggere il repository. CodeCommit
Un errore di concorrenza, ad esempio. `Project-level concurrent build limit cannot exceed the account-l evel concurrent build limit of 1`	Esegui nuovamente la pipeline scegliendo il pulsante Release Change nella console. CodePipeline Si tratta di un problema noto che sembra essere più comune durante le prime fasi di esecuzione della pipeline.

Risorse correlate

Fornisci feedback sul progetto SCSP.

Informazioni aggiuntive

DOMANDE FREQUENTI

Il progetto SCSP è lo stesso di Automated Security Helper (ASH)?

No. Usa ASH quando desideri uno strumento CLI che esegua strumenti di scansione del codice utilizzando contenitori. Automated Security Helper (ASH) è uno strumento progettato per ridurre la probabilità di una violazione della sicurezza nella nuova configurazione di codice, infrastruttura o risorsa IAM. ASH è un'utilità da riga di comando che può essere eseguita localmente. L'uso locale richiede l'installazione e il funzionamento di un ambiente contenitore sul sistema.

Usa SCSP quando desideri una pipeline di configurazione più semplice rispetto a ASH. SCSP non richiede installazioni locali. SCSP è progettato per eseguire i controlli singolarmente in una pipeline e visualizzare i risultati per strumento. SCSP evita inoltre un notevole sovraccarico di lavoro legato alla configurazione di Docker ed è indipendente dal sistema operativo (OS).

SCSP è solo per i team di sicurezza?

No, chiunque può implementare la pipeline per determinare quali parti del proprio codice non superano i controlli di sicurezza. Ad esempio, gli utenti che non si occupano di sicurezza possono utilizzare SCSP per verificare il codice prima di esaminarlo con i propri team di sicurezza.

Posso usare SCSP se lavoro con un altro tipo di repository, ad esempio, o Bitbucket GitLab? GitHub

Puoi configurare un repository git locale in modo che punti a due diversi repository remoti. Ad esempio, è possibile clonare un GitLab repository esistente, creare un'istanza SCSP (specificando CloudFormation, se necessario, le cartelle Terraform e AWS Config Rules Development Kit (AWS RDK)) e quindi utilizzarla anche per indirizzare l'archivio locale `git remote add upstream <SCSPGitLink>` verso l'archivio SCSP. CodeCommit Ciò consente di inviare prima le modifiche al codice a SCSP, di convalidarle e quindi, dopo eventuali aggiornamenti aggiuntivi per correggere i risultati, di GitLab inviarle GitHub al repository o Bitbucket. Per ulteriori informazioni sui telecomandi multipli, vedi Inviare i commit a un repository Git aggiuntivo (AWS post del blog).

> **Note**
>
> Fai attenzione alle deviazioni, ad esempio evita di apportare modifiche tramite interfacce web.

Contribuisci e aggiungi le tue azioni

La configurazione di SCSP viene gestita come GitHub progetto, che contiene il codice sorgente per l'applicazione SCSP AWS Cloud Development Kit (AWS CDK) . Per aggiungere ulteriori controlli alla pipeline, l' AWS CDK applicazione deve essere aggiornata e quindi sintetizzata o distribuita nella

destinazione Account AWS in cui verrà eseguita la pipeline. Per fare ciò, inizia clonando il GitHub progetto SCSP, quindi trova il file di definizione dello stack nella cartella. `lib`

Se desideri aggiungere un controllo aggiuntivo, la `StandardizedCodeBuildProject` classe nel AWS CDK codice semplifica l'aggiunta di azioni. Fornisci il nome, la descrizione `install` e/o `build` i comandi. AWS CDK crea il CodeBuild progetto utilizzando valori predefiniti ragionevoli. Oltre a creare il progetto di compilazione, è necessario aggiungerlo alle CodePipeline azioni in fase di compilazione. Quando si progetta un nuovo controllo, l'azione dovrebbe essere eseguita `FAIL` se lo strumento di scansione rileva problemi o non riesce a funzionare. L'azione dovrebbe avvenire `PASS` se lo strumento di scansione non rileva alcun problema. Per un esempio di configurazione di uno strumento, consulta il codice dell'`Bandit`azione.

Per ulteriori informazioni sugli input e sugli output previsti, consulta la documentazione del repository.

Se si aggiungono azioni personalizzate, è necessario distribuire SCSP utilizzando o. `cdk deploy` `cdk synth + CloudFormation deploy` Questo perché il CloudFormation modello Quick create stack è gestito dai proprietari del repository.

Implementa controlli di accesso basati su attributi investigativi per sottoreti pubbliche utilizzando AWS Config

Alberto Menéndez, Amazon Web Services

Riepilogo

Le architetture di rete periferiche distribuite si basano sulla sicurezza perimetrale della rete che funziona insieme ai carichi di lavoro nei loro cloud privati virtuali (). VPCs Ciò offre una scalabilità senza precedenti rispetto all'approccio centralizzato più comune. Sebbene l'implementazione di sottoreti pubbliche negli account di carico di lavoro possa offrire vantaggi, introduce anche nuovi rischi per la sicurezza perché aumenta la superficie di attacco. Si consiglia di distribuire solo risorse Elastic Load Balancing, come Application Load Balancer o gateway NAT nelle relative sottoreti pubbliche. VPCs L'utilizzo di sistemi di bilanciamento del carico e gateway NAT in sottoreti pubbliche dedicate consente di implementare un controllo granulare del traffico in entrata e in uscita.

Ti consigliamo di implementare controlli sia preventivi che investigativi per limitare i tipi di risorse che possono essere distribuite nelle sottoreti pubbliche. Per ulteriori informazioni sull'utilizzo del controllo degli accessi basato sugli attributi (ABAC) per implementare controlli preventivi per le sottoreti pubbliche, consulta Implementare controlli di accesso preventivi basati sugli attributi per le sottoreti pubbliche. Sebbene efficaci per la maggior parte delle situazioni, questi controlli preventivi potrebbero non risolvere tutti i possibili casi d'uso. Pertanto, questo modello si basa sull'approccio ABAC e consente di configurare gli avvisi sulle risorse non conformi distribuite nelle sottoreti pubbliche. La soluzione verifica se le interfacce di rete elastiche appartengono a una risorsa non consentita nelle sottoreti pubbliche.

A tal fine, questo modello utilizza regole AWS Config personalizzate e ABAC. La regola personalizzata elabora la configurazione di un'interfaccia di rete elastica ogni volta che viene creata o modificata. A un livello elevato, questa regola esegue due azioni per determinare se l'interfaccia di rete è conforme:

1. Per determinare se l'interfaccia di rete rientra nell'ambito della regola, la regola verifica se la sottorete ha AWS tag specifici che indicano che si tratta di una sottorete pubblica. Ad esempio, questo tag potrebbe essere. `IsPublicFacing=True`
2. Se l'interfaccia di rete è distribuita in una sottorete pubblica, la regola verifica chi Servizio AWS ha creato questa risorsa. Se la risorsa non è una risorsa Elastic Load Balancing o un gateway NAT, la contrassegna come non conforme.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- AWS Config, configurato nell'account del carico di lavoro
- Autorizzazioni per distribuire le risorse richieste nell'account del carico di lavoro
- Un VPC con sottoreti pubbliche
- Tag applicati correttamente per identificare le sottoreti pubbliche di destinazione
- (Facoltativo) Un'organizzazione in AWS Organizations
- (Facoltativo) Un account di sicurezza centrale che funge da amministratore delegato per AWS Config e AWS Security Hub CSPM

Architecture

Architettura Target

Il diagramma illustra quanto segue:

1. Quando una risorsa di elastic network interface (`AWS::EC2::NetworkInterface`) viene distribuita o modificata, AWS Config acquisisce l'evento e la configurazione.
2. AWS Config confronta questo evento con la regola personalizzata utilizzata per valutare la configurazione.
3. Viene richiamata la AWS Lambda funzione associata a questa regola personalizzata. La funzione valuta la risorsa e applica la logica specificata per determinare se la configurazione della risorsa è`COMPLIANT`, `NON_COMPLIANT` o. `NOT_APPLICABLE`
4. Se si determina che una risorsa lo è`NON_COMPLIANT`, AWS Config invia un avviso tramite Amazon Simple Notification Service (Amazon SNS).

> Nota: se questo account è un account membro di AWS Organizations, puoi inviare i dati di conformità a un account di sicurezza centrale tramite AWS Config o AWS Security Hub CSPM.

Logica di valutazione della funzione Lambda

Il diagramma seguente mostra la logica applicata dalla funzione Lambda per valutare la conformità dell'elastic network interface.

Automazione e scalabilità

Questo schema è una soluzione investigativa. È inoltre possibile integrarlo con una regola di correzione per risolvere automaticamente le risorse non conformi. Per ulteriori informazioni, vedere Risanamento delle risorse non conformi con le regole. AWS Config

È possibile scalare questa soluzione in base a:

- Imposizione dell'applicazione dei AWS tag corrispondenti stabiliti per identificare le sottoreti rivolte al pubblico. Per ulteriori informazioni, consulta le politiche relative ai tag nella documentazione. AWS Organizations
- Configurazione di un account di sicurezza centralizzato che applichi la regola AWS Config personalizzata a ogni account di carico di lavoro dell'organizzazione. Per ulteriori informazioni, consulta Automatizzare la conformità alla configurazione su larga scala in AWS (AWS post del blog).
- Integrazione AWS Config con AWS Security Hub CSPM per acquisire, centralizzare e notificare su larga scala. Per ulteriori informazioni, consulta Configurazione AWS Config nella documentazione. AWS Security Hub CSPM

Tools (Strumenti)

- AWS Configfornisce una visualizzazione dettagliata delle risorse presenti Account AWS e di come sono configurate. Ti aiuta a identificare in che modo le risorse sono correlate tra loro e in che modo le loro configurazioni sono cambiate nel tempo.
- Elastic Load Balancing distribuisce il traffico di applicazioni o di rete in entrata su più destinazioni. Ad esempio, puoi distribuire il traffico tra istanze Amazon Elastic Compute Cloud (Amazon EC2), contenitori e indirizzi IP in una o più zone di disponibilità.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.

- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.

Best practice

Per altri esempi e best practice per lo sviluppo di AWS Config regole personalizzate, consulta il AWS Config Rules Repository ufficiale su. GitHub

Epiche

Implementazione della soluzione

Operazione	Description	Competenze richieste
Creazione della funzione Lambda	1. Accedere a Console di gestione AWS, quindi aprire la AWS Lambda console. 2. Nella pagina Functions (Funzioni), scegli Create function (Crea funzione). 3. Scegli Crea da zero. 4. Nel riquadro Informazioni di base, per Nome della funzione, inserisci un nome. 5. In Runtime, scegli Python 3.12. 6. Lascia l'architettura impostata su x86_64. 7. Scegli Crea funzione. 8. Scegli la scheda Codice. 9. In Esplora file, scegli lambda_function.py. 10. Incolla il codice di esempio fornito nella sezione	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	Informazioni aggiuntive di questo modello nella scheda lambda_function.py. Personalizzate il codice di esempio per identificare qualsiasi logica di valutazione personalizzata nella `evaluate_change_notification_compliance` funzione. 11. Seleziona Implementa.	

Operazione	Description	Competenze richieste
Aggiungi le autorizzazioni al ruolo di esecuzione della funzione Lambda.	1. Nel riquadro di navigazione, seleziona Funzioni. 2. Scegli la funzione che hai appena creato. 3. Scegli Configuration (Configurazione), quindi Permissions (Autorizz azioni). 4. Scegli il nome del ruolo per aprirlo nella console AWS Identity and Access Management (IAM). 5. In Politiche di autorizza zione, scegli Aggiungi autorizzazioni, quindi scegli Crea politica in linea. 6. Scegli JSON. 7. Incolla la seguente politica nell'editor delle politiche. Ciò consente alla funzione Lambda di: • Ottieni i dettagli dei tag della sottorete. • Invia il risultato della conformità a. AWS Config `{ "Version": "2012-10-17", "Statement": [ { "Action": [`	Informazioni generali su AWS

Operazione	Description	Competenze richieste

	<pre>"config:PutEvaluations",
	"ec2:DescribeSubnets"
	],
	"Resource": "*",
	"Effect": "Allow"
	}
	]
	}</pre>
	8. Scegli Next (Successivo).
	9. Immettere un nome per la policy e scegliere Create policy (Crea policy).	

Recupera la funzione Lambda Amazon Resource Name (ARN).	1. Aprire la console Lambda. 2. Nel riquadro di navigazione, seleziona Funzioni. 3. Scegli la funzione che hai appena creato. 4. Nella sezione Panoramica delle funzioni, in Funzione ARN, copia il valore.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Crea la regola AWS Config personalizzata.	1. Apri la AWS Config console. 2. Nella pagina Rules (Regole) scegli Add rule (Aggiungi regola). 3. Nella pagina Specificare il tipo di regola, scegli Crea regola Lambda personali zzata, quindi scegli Avanti. 4. Nella pagina Configura regola, procedi come segue: a. Inserisci un nome e una descrizione. b. Per la AWS Lambda funzione ARN, incolla l'ARN che hai copiato in precedenza. c. In Tipo di trigger, scegli Quando cambia la configurazione. d. Per Ambito delle modifiche, seleziona Risorse. e. Per Tipo di risorsa, scegli AWS EC2 NetworkIn terface. f. Scegli Next (Successi vo). 5. Nella pagina Rivedi e crea, verifica la regola, quindi scegli Salva.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Configura le notifiche.	1. Segui le istruzioni in Creazione di un argomento Amazon SNS per creare un argomento Amazon SNS. 2. Segui le istruzioni nell'argomento Abbonamento a un Amazon SNS per configurare un endpoint che riceve notifiche per l'argomento Amazon SNS. 3. Segui le istruzioni in Come posso ricevere una notifica quando una AWS risorsa non è conforme? Utilizza AWS Config per configurare una EventBridge regola Amazon personalizzata per le tue risorse non conformi.	Informazioni generali su AWS

Test della soluzione

Operazione	Description	Competenze richieste
Crea una risorsa conforme.	1. Utilizza le seguenti istruzioni per creare una delle risorse supportate in una sottorete pubblica: • Creare un gateway NAT • Getting started with Network Load Balancers • Creare un Application Load Balancer	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	2. Dopo la creazione della risorsa, la regola AWS Config personalizzata valuta le interfacce di rete elastiche associate alla risorsa. Contrassegna queste interfacce di rete come. `COMPLIANT` È possibile visualizzare le risorse in AWS Config seguendo questi passaggi: a. Apri la AWS Config console. b. Nella pagina Regole, scegli la tua regola. c. Nella pagina dei dettagli della regola, vai in fondo alla pagina. d. In Risorse nell'ambito, seleziona Conforme. Conferma di vedere le interfacce IDs di rete che sono state create. e. Per maggiori dettagli sulla configurazione dell'interfaccia di rete, scegli l'ID della risorsa.	

Operazione	Description	Competenze richieste
Crea una risorsa non conforme.	1. Utilizza le seguenti istruzioni per creare una risorsa non conforme in una sottorete pubblica: • Avvia un' EC2 istanza Amazon • Creazione di un'istanza di database Amazon Relational Database Service (Amazon RDS) • Crea un endpoint VPC 2. Dopo la creazione della risorsa, la regola AWS Config personalizzata valuta le interfacce di rete elastiche associate alla risorsa. Contrassegna queste interfacce di rete come. `NON_COMPLIANT` È possibile visualizzare le risorse in AWS Config seguendo questi passaggi: a. Apri la AWS Config console. b. Nella pagina Regole, scegli la tua regola. c. Nella pagina dei dettagli della regola, vai in fondo alla pagina. d. In Risorse nell'ambito, seleziona NonCompliant. Conferma di vedere le	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	interfacce IDs di rete che sono state create. e. Per maggiori dettagli sulla configurazione dell'interfaccia di rete, scegli l'ID della risorsa. 3. Conferma di aver ricevuto la notifica sull'endpoint che hai configurato in Amazon SNS.	
Crea una risorsa non applicabile.	1. In una sottorete privata, create qualsiasi risorsa che richieda un'interfaccia di rete elastica. 2. Dopo la creazione della risorsa, la regola AWS Config personalizzata valuta le interfacce di rete elastiche associate alla risorsa. Contrassegna queste interfacce di rete come. `NOT_APPLICABLE` Queste risorse non vengono visualizzate nella AWS Config console.	Informazioni generali su AWS

Risorse correlate

AWS documentazione

- Configurazione AWS Config
- AWS Config regole personalizzate
- ABAC per AWS

- Implementa controlli di accesso preventivi basati sugli attributi per le sottoreti pubbliche

AWS Altre risorse

- Automatizza la conformità alla configurazione su larga scala in AWS
- Architetture di ispezione distribuite con Gateway Load Balancer

Informazioni aggiuntive

Di seguito è riportato un esempio di funzione Lambda fornita a scopo dimostrativo.

```
import boto3
import json
import os

# Init clients
config_client = boto3.client('config')
ec2_client = boto3.client('ec2')

def lambda_handler(event, context):

    # Init values
    compliance_value = 'NOT_APPLICABLE'
    invoking_event = json.loads(event['invokingEvent'])
    configuration_item = invoking_event['configurationItem']

    status = configuration_item['configurationItemStatus']
    eventLeftScope = event['eventLeftScope']

    # First check if the event configuration applies. Ex. resource event is not delete
    if (status == 'OK' or status == 'ResourceDiscovered') and not eventLeftScope:
        compliance_value = evaluate_change_notification_compliance(configuration_item)


    config_client.put_evaluations(
       Evaluations=[
           {
               'ComplianceResourceType': invoking_event['configurationItem']
['resourceType'],
               'ComplianceResourceId': invoking_event['configurationItem']
['resourceId'],
```

```
                'ComplianceType': compliance_value,
                'OrderingTimestamp': invoking_event['configurationItem']
['configurationItemCaptureTime']
            },
        ],
        ResultToken=event['resultToken'])

# Function with the logs to evaluate the resource
def evaluate_change_notification_compliance(configuration_item):
    is_in_scope = is_in_scope_subnet(configuration_item['configuration']['subnetId'])

    if (configuration_item['resourceType'] != 'AWS::EC2::NetworkInterface') or not
 is_in_scope:
        return 'NOT_APPLICABLE'

    else:
        alb_condition = configuration_item['configuration']['requesterId'] in ['amazon-
elb']
        nlb_condition = configuration_item['configuration']['interfaceType'] in
 ['network_load_balancer']
        nat_gateway_condition = configuration_item['configuration']['interfaceType'] in
 ['nat_gateway']

        if alb_condition or nlb_condition or nat_gateway_condition:
            return 'COMPLIANT'
    return 'NON_COMPLIANT'

# Function to check if elastic network interface is in public subnet
def is_in_scope_subnet(eni_subnet):

    subnet_description = ec2_client.describe_subnets(
        SubnetIds=[eni_subnet]
    )

    for subnet in subnet_description['Subnets']:
        for tag in subnet['Tags']:
            if tag['Key'] == os.environ.get('TAG_KEY') and tag['Value'] ==
 os.environ.get('TAG_VALUE'):
                return True

    return False
```

Implementa controlli di accesso preventivi basati sugli attributi per le sottoreti pubbliche

Joel Alfredo Núñez González e Samuel Ortega Sancho, Amazon Web Services

Riepilogo

Nelle architetture di rete centralizzate, i cloud privati virtuali di ispezione ed edge (VPCs) concentrano tutto il traffico in entrata e in uscita, ad esempio il traffico da e verso Internet. Tuttavia, ciò può creare colli di bottiglia o portare al raggiungimento dei limiti delle quote di servizio AWS. L'implementazione della sicurezza perimetrale della rete insieme ai carichi di lavoro previsti VPCs offre una scalabilità senza precedenti rispetto all'approccio più comune e centralizzato. Questa è chiamata architettura perimetrale distribuita.

Sebbene l'implementazione di sottoreti pubbliche negli account di carico di lavoro possa offrire vantaggi, introduce anche nuovi rischi per la sicurezza perché aumenta la superficie di attacco. Si consiglia di distribuire solo risorse Elastic Load Balancing (ELB), come Application Load Balancer o gateway NAT nelle relative sottoreti pubbliche. VPCs L'utilizzo di sistemi di bilanciamento del carico e gateway NAT in sottoreti pubbliche dedicate consente di implementare un controllo granulare del traffico in entrata e in uscita.

Il controllo degli accessi basato sugli attributi (ABAC) è la pratica di creare autorizzazioni dettagliate basate su attributi utente, come reparto, ruolo professionale e nome del team. Per ulteriori informazioni, consulta ABAC for AWS. ABAC può fornire guardrail per sottoreti pubbliche negli account di carico di lavoro. Questo aiuta i team applicativi a essere agili, senza compromettere la sicurezza dell'infrastruttura.

Questo modello descrive come contribuire a proteggere le sottoreti pubbliche implementando ABAC tramite una policy di controllo dei servizi (SCP) in AWS Organizations e policy in AWS Identity and Access Management (IAM). Puoi applicare l'SCP a un account membro di un'organizzazione o a un'unità organizzativa (OU). Queste politiche ABAC consentono agli utenti di implementare gateway NAT nelle sottoreti di destinazione e impediscono loro di distribuire altre risorse Amazon Elastic Compute Cloud (Amazon EC2), come istanze e interfacce di rete elastiche. EC2

Prerequisiti e limitazioni

Prerequisiti

- Un'organizzazione in AWS Organizations
- Accesso amministrativo all'account root di AWS Organizations
- Nell'organizzazione, un account membro attivo o un'unità organizzativa per testare l'SCP

Limitazioni

- L'SCP di questa soluzione non impedisce ai servizi AWS che utilizzano un ruolo collegato ai servizi di distribuire risorse nelle sottoreti di destinazione. Esempi di questi servizi sono Elastic Load Balancing (ELB), Amazon Elastic Container Service (Amazon ECS) e Amazon Relational Database Service (Amazon RDS). Per ulteriori informazioni, consulta gli effetti SCP sulle autorizzazioni nella documentazione di AWS Organizations. Implementa controlli di sicurezza per rilevare queste eccezioni.

Architecture

Stack tecnologico Target

- SCP applicato a un account AWS o a un'unità organizzativa in AWS Organizations
- I seguenti ruoli IAM:
  - `AutomationAdminRole`— Utilizzato per modificare i tag di sottorete e creare risorse VPC dopo l'implementazione di SCP
  - `TestAdminRole`— Utilizzato per verificare se SCP impedisce ad altri principali IAM, compresi quelli con accesso amministrativo, di eseguire le azioni riservate a `AutomationAdminRole`

Architettura Target

1. Crei il ruolo `AutomationAdminRole` IAM nell'account di destinazione. Questo ruolo dispone delle autorizzazioni per gestire le risorse di rete. Nota le seguenti autorizzazioni che sono esclusive per questo ruolo:
   - Questo ruolo può creare VPCs sottoreti pubbliche.
   - Questo ruolo può modificare le assegnazioni dei tag per le sottoreti di destinazione.
   - Questo ruolo può gestire le proprie autorizzazioni.
2. In AWS Organizations, applichi l'SCP all'account AWS o all'unità organizzativa di destinazione. Per un esempio di policy, consulta Informazioni aggiuntive in questo modello.

3. Un utente o uno strumento nella CI/CD pipeline può assumere il `AutomationAdminRole` ruolo di applicare il `SubnetType` tag alle sottoreti di destinazione.
4. Assumendo altri ruoli IAM, i responsabili IAM autorizzati dell'organizzazione possono gestire i gateway NAT nelle sottoreti di destinazione e altre risorse di rete consentite nell'account AWS, come le tabelle di routing. Utilizza le policy IAM per concedere queste autorizzazioni. Per ulteriori informazioni, consulta Gestione delle identità e degli accessi per Amazon VPC.

Automazione e scalabilità

Per proteggere le sottoreti pubbliche, è necessario applicare i tag AWS corrispondenti. Dopo aver applicato l'SCP, i gateway NAT sono l'unico tipo di EC2 risorsa Amazon che gli utenti autorizzati possono creare nelle sottoreti dotate del tag. `SubnetType:IFA` (significa risorse rivolte a Internet.) *`IFA`* L'SCP impedisce la creazione di altre EC2 risorse Amazon, come istanze e interfacce di rete elastiche. Ti consigliamo di utilizzare una CI/CD pipeline che assuma il `AutomationAdminRole` ruolo di creare risorse VPC in modo che questi tag vengano applicati correttamente alle sottoreti pubbliche.

Tools (Strumenti)

Servizi AWS

- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Organizations è un servizio di gestione degli account che ti aiuta a consolidare più account AWS in un'organizzazione da creare e gestire centralmente. In AWS Organizations, puoi implementare le policy di controllo dei servizi (SCPs), che sono un tipo di policy che puoi utilizzare per gestire le autorizzazioni nella tua organizzazione.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare le risorse AWS in una rete virtuale che hai definito. Questa rete virtuale è simile a una rete tradizionale che gestiresti nel tuo data center, con i vantaggi dell'utilizzo dell'infrastruttura scalabile di AWS.

Epiche

Applica l'SCP

Operazione	Description	Competenze richieste
Crea un ruolo di amministratore di test.	Crea un ruolo IAM denominato `TestAdminRole` nell'account AWS di destinazione. Allega la policy AdministratorAccessAWS managed IAM al nuovo ruolo. Per istruzioni, consulta Creazione di un ruolo per delegare le autorizzazioni a un utente IAM nella documentazione IAM.	Amministratore AWS
Crea il ruolo di amministratore dell'automazione.	1. Crea un ruolo IAM denominato `AutomationAdminRole` nell'account AWS di destinazione. 2. Allega la policy AdministratorAccessAWS managed IAM al nuovo ruolo. Di seguito è riportato un esempio di policy di fiducia che è possibile utilizzare per testare il ruolo dall'`111122223333` account. `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow",`	Amministratore AWS

Operazione	Description	Competenze richieste

	<pre>"Principa
	l": {
	"AWS": [
	
	"arn:aws:iam::1111
	22223333:root"
	]
	},
	"Action":
	"sts:AssumeRole",
	"Conditio
	n": {}
	}
	]
	}</pre>	

Crea e collega l'SCP.	1. Utilizzando il codice di esempio fornito nella sezione Informazioni aggiuntive, create una politica di controllo della sicurezza. Per istruzioni, consulta Creazione di un SCP nella documentazione di AWS Organizations. 2. Collega l'SCP all'account AWS o all'unità organizzativa di destinazione. Per istruzioni, consulta Allegare e scollegare le policy di controllo del servizio nella documentazione di AWS Organizations.	Amministratore AWS

Prova l'SCP

Operazione	Description	Competenze richieste
Crea un VPC o una sottorete.	1. Assumi il `TestAdmin Role` ruolo nell'account AWS di destinazione. 2. Prova a creare un VPC o una nuova sottorete pubblica in un VPC esistente. Per istruzioni, consulta Creare un VPC, sottoreti e altre risorse VPC nella documentazione di Amazon VPC. Non dovresti essere in grado di creare queste risorse. 3. Assumi il `Automatio nAdminRole` ruolo e riprova il passaggio precedente. Ora dovresti essere in grado di creare le risorse di rete.	Amministratore AWS
Gestisci i tag.	1. Assumi il `TestAdmin Role` ruolo nell'account AWS di destinazione. 2. Aggiungi un `SubnetTyp e:IFA` tag a una sottorete pubblica disponibile. Dovresti essere in grado di aggiungere questo tag. Per istruzioni su come aggiungere tag tramite l'AWS Command Line Interface (AWS CLI),	Amministratore AWS

Operazione	Description	Competenze richieste
	consulta create-tags nell'AWS CLI Command Reference. 3. Senza modificare le credenziali, prova a modificare il `SubnetType:IFA` tag assegnato a questa sottorete. Non dovresti essere in grado di modificare questo tag. 4. Assumi il `AutomationAdminRole` ruolo e riprova i passaggi precedenti. Questo ruolo dovrebbe essere in grado di aggiungere e modificare questo tag.	

Operazione	Description	Competenze richieste
Distribuisci risorse nelle sottoreti di destinazione.	1. Assumi il ruolo. `TestAdmin Role` 2. Per una sottorete pubblica con il `SubnetType:IFA` tag, provate a creare un' EC2 istanza. Per istruzioni, consulta Launch an istance nella EC2 documenta zione di Amazon. In questa sottorete, non dovresti essere in grado di creare, modificare o eliminare alcuna EC2 risorsa Amazon ad eccezione dei gateway NAT. 3. Nella stessa sottorete, crea un gateway NAT. Per istruzioni, consulta Creare un gateway NAT nella documentazione di Amazon VPC. Dovresti essere in grado di creare, modificare o eliminare i gateway NAT in questa sottorete.	Amministratore AWS

Operazione	Description	Competenze richieste
Gestisci il AutomationAdminRole ruolo.	1. Assumi il `TestAdminRole` ruolo. 2. Prova a modificare il `AutomationAdminRole` ruolo. Per istruzioni, consulta Modifica di un ruolo nella documentazione IAM. Non dovresti essere in grado di modificare questo ruolo. 3. Assumi il `AutomationAdminRole` ruolo e riprova il passaggio precedente. Ora dovresti essere in grado di modificare il ruolo.	Amministratore AWS

Eliminazione

Operazione	Description	Competenze richieste
Pulisci le risorse distribuite.	1. Scollega SCP dall'account AWS o dall'unità organizzativa. Per istruzioni, consulta Detaching an SCP nella documentazione di AWS Organizations. 2. Eliminare l'SCP. Per istruzioni, consulta Eliminazione di un SCP (documentazione di AWS Organizations).	Amministratore AWS

Operazione	Description	Competenze richieste
	3. Elimina il `AutomationAdminRole` ruolo e il `TestAdminRole` ruolo. Per istruzioni, consulta Eliminazione dei ruoli nella documentazione IAM. 4. Elimina tutte le risorse di rete, ad esempio VPCs e le sottoreti, che hai creato per questa soluzione.	

Risorse correlate

Documentazione AWS

- Collegamento e scollegamento SCPs
- Creazione, aggiornamento ed eliminazione SCPs
- Implementa controlli di accesso basati su attributi investigativi per sottoreti pubbliche utilizzando AWS Config
- Controlli di rilevamento
- Riferimento per l'autorizzazione del servizio
- Assegnazione di tag alle risorse AWS
- Cos'è ABAC per AWS?

Riferimenti AWS aggiuntivi

- Protezione dei tag delle risorse utilizzati per l'autorizzazione utilizzando una Service Control Policy in AWS Organizations (post sul blog AWS)

Informazioni aggiuntive

La seguente policy di controllo dei servizi è un esempio che puoi usare per testare questo approccio nella tua organizzazione.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyVPCActions",
      "Effect": "Deny",
      "Action": [
        "ec2:CreateVPC",
        "ec2:CreateRoute",
        "ec2:CreateSubnet",
        "ec2:CreateInternetGateway",
        "ec2:DeleteVPC",
        "ec2:DeleteRoute",
        "ec2:DeleteSubnet",
        "ec2:DeleteInternetGateway"
      ],
      "Resource": [
        "arn:aws:ec2:*:*:*"
      ],
      "Condition": {
        "StringNotLike": {
          "aws:PrincipalARN": ["arn:aws:iam::*:role/AutomationAdminRole"]
        }
      }
    },
    {
      "Sid": "AllowNATGWOnIFASubnet",
      "Effect": "Deny",
      "NotAction": [
        "ec2:CreateNatGateway",
        "ec2:DeleteNatGateway"
      ],
      "Resource": [
        "arn:aws:ec2:*:*:subnet/*"
      ],
      "Condition": {
        "ForAnyValue:StringEqualsIfExists": {
          "aws:ResourceTag/SubnetType": "IFA"
        },
        "StringNotLike": {
          "aws:PrincipalARN": ["arn:aws:iam::*:role/AutomationAdminRole"]
        }
      }
```

```
    },
    {
      "Sid": "DenyChangesToAdminRole",
      "Effect": "Deny",
      "NotAction": [
        "iam:GetContextKeysForPrincipalPolicy",
        "iam:GetRole",
        "iam:GetRolePolicy",
        "iam:ListAttachedRolePolicies",
        "iam:ListInstanceProfilesForRole",
        "iam:ListRolePolicies",
        "iam:ListRoleTags"
      ],
      "Resource": [
        "arn:aws:iam::*:role/AutomationAdminRole"
      ],
      "Condition": {
        "StringNotLike": {
          "aws:PrincipalARN": ["arn:aws:iam::*:role/AutomationAdminRole"]
        }
      }
    }
  ]
}
```

Rileva le istanze di database Amazon RDS e Aurora con certificati CA in scadenza

Stephen DiCato ed Eugene Shifer, Amazon Web Services

Riepilogo

Come best practice di sicurezza, si consiglia di crittografare i dati in transito tra i server delle applicazioni e i database relazionali. È possibile utilizzare SSL o TLS per crittografare una connessione a un'istanza o un cluster di database (DB). Questi protocolli aiutano a garantire riservatezza, integrità e autenticità tra un'applicazione e un database. Il database utilizza un certificato server, rilasciato da un'autorità di certificazione (CA) e utilizzato per eseguire la verifica dell'identità del server. SSL o TLS verifica l'autenticità del certificato convalidandone la firma digitale e assicurandosi che non sia scaduto.

Nel Console di gestione AWS, Amazon Relational Database Service (Amazon RDS) e Amazon Aurora forniscono notifiche sulle istanze DB che richiedono aggiornamenti dei certificati. Tuttavia, per verificare la presenza di queste notifiche, è necessario accedere a ciascuna di esse Account AWS e accedere alla console di servizio di ciascuna di esse. Regione AWS Questa attività diventa più complessa se è necessario valutare la validità dei certificati per molti Account AWS certificati gestiti come organizzazione in AWS Organizations.

Effettuando il provisioning dell'infrastruttura come codice (IaC) fornita in questo schema, puoi rilevare i certificati CA in scadenza per tutte le istanze DB Amazon RDS e Aurora della tua organizzazione. Account AWS AWS Il AWS CloudFormationmodello fornisce tutte le AWS Config regole, le funzioni e le autorizzazioni necessarie AWS Lambda . È possibile distribuirlo in un singolo account come stack oppure distribuirlo in tutta l' AWS organizzazione come set di stack.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Se stai distribuendo in un'unica Account AWS soluzione:
  - Assicurati di disporre delle autorizzazioni per creare CloudFormation pile.
  - Abilita AWS Config nell'account di destinazione.
  - (Facoltativo) Abilita AWS Security Hub CSPM nell'account di destinazione.

- Se ti stai distribuendo in un' AWS organizzazione:
  - Assicurati di disporre delle autorizzazioni per creare set di CloudFormation stack.
  - Abilita Security Hub CSPM con AWS Organizations integrazione.
  - Abilita questa soluzione AWS Config negli account in cui stai implementando questa soluzione.
  - Account AWS Designare un amministratore delegato per AWS Config Sand Security Hub CSPM.

Limitazioni

- Se esegui la distribuzione su un account individuale che non ha il CSPM di Security Hub abilitato, puoi utilizzarlo AWS Config per valutare i risultati.
- Se stai eseguendo la distribuzione in un'organizzazione che non dispone di un amministratore delegato per AWS Config Security Hub CSPM, devi accedere agli account dei singoli membri per visualizzare i risultati.
- Se lo utilizzi AWS Control Tower per gestire e governare gli account della tua organizzazione, distribuisci IaC secondo questo schema utilizzando Customizations for (cFCT). AWS Control Tower L'utilizzo della CloudFormation console creerà differenze di configurazione e richiederà la nuova AWS Control Tower registrazione delle unità organizzative () o degli account gestiti. OUs
- Alcuni Servizi AWS non sono disponibili in tutti. Regioni AWS Per informazioni sulla disponibilità regionale, consulta la pagina Endpoint e quote del servizio e scegli il link relativo al servizio.

Architecture

Implementazione in un individuo Account AWS

Il seguente diagramma di architettura mostra l'implementazione delle AWS risorse all'interno di una singola unità. Account AWS Viene implementato utilizzando un CloudFormation modello direttamente tramite la CloudFormation console. Se Security Hub CSPM è abilitato, è possibile visualizzare i risultati in uno AWS Config o in Security Hub CSPM. Se Security Hub CSPM non è abilitato, è possibile visualizzare i risultati solo nella AWS Config console.

Il diagramma mostra i seguenti passaggi:

1. Crei una CloudFormation pila. Questo implementa una funzione Lambda e AWS Config una regola. Sia la regola che la funzione sono configurate con le autorizzazioni AWS Identity and

Access Management (IAM) necessarie per pubblicare le valutazioni delle risorse nei registri e nei AWS Config log.

2. La AWS Config regola funziona in modalità di valutazione investigativa e viene eseguita ogni 24 ore.
3. Security Hub CSPM riceve tutti i AWS Config risultati.
4. È possibile visualizzare i risultati in Security Hub CSPM o in AWS Config, a seconda della configurazione dell'account.

Implementazione in un'organizzazione AWS

Il diagramma seguente mostra la valutazione della scadenza dei certificati su più account gestiti tramite AWS Organizations e. AWS Control Tower Il CloudFormation modello viene distribuito tramite cFCT. I risultati della valutazione sono centralizzati in Security Hub CSPM nell'account amministratore delegato. Il AWS CodePipeline flusso di lavoro illustrato nel diagramma mostra i passaggi di base che si verificano durante l'implementazione di cFCT.

Il diagramma mostra i seguenti passaggi:

1. A seconda della configurazione per cFct, nell'account di gestione, si invia l'IaC a un AWS CodeCommit repository o si carica un file compresso (ZIP) dell'IaC in un bucket Amazon Simple Storage Service (Amazon S3).
2. La pipeline cFct decomprime il file, esegue i controlli cfn-nag () e lo distribuisce come set di stack. GitHub CloudFormation
3. A seconda della configurazione specificata nel file manifest cFct, distribuisce gli stack in account individuali o specificati. CloudFormation StackSets OUs Questo implementa una funzione Lambda e AWS Config una regola negli account di destinazione. Sia la regola che la funzione sono configurate con le autorizzazioni IAM necessarie per pubblicare le valutazioni delle risorse nei log e nei AWS Config log.
4. La AWS Config regola funziona in modalità di valutazione investigativa e viene eseguita ogni 24 ore.
5. AWS Config inoltra tutti i risultati a Security Hub CSPM.
6. I risultati CSPM di Security Hub vengono aggregati nell'account amministratore delegato.
7. È possibile visualizzare i risultati in Security Hub CSPM nell'account amministratore delegato.

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationti aiuta a configurare AWS le risorse, a fornirle in modo rapido e coerente e a gestirle durante tutto il loro ciclo di vita in tutte le regioni. Account AWS
- AWS Configfornisce una visione dettagliata delle risorse disponibili Account AWS e di come sono configurate. Ti aiuta a identificare in che modo le risorse sono correlate tra loro e in che modo le loro configurazioni sono cambiate nel tempo. Una AWS Config regola definisce le impostazioni di configurazione ideali per una risorsa e consente di AWS Config valutare se le AWS risorse soddisfano le condizioni stabilite dalle regole.
- AWS Control Towerti aiuta a configurare e gestire un ambiente con AWS più account, seguendo le migliori pratiche prescrittive. Customizations for AWS Control Tower (cFct) ti aiuta a personalizzare la tua AWS Control Tower landing zone e a rimanere in linea con AWS le migliori pratiche. Le personalizzazioni sono implementate con CloudFormation modelli e politiche di controllo del servizio (). SCPs
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS Organizationsè un servizio di gestione degli account che ti aiuta a consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente.
- AWS Security Hub CSPMfornisce una visione completa dello stato di sicurezza in AWS. Inoltre, consente di verificare la conformità AWS dell'ambiente agli standard e alle best practice del settore della sicurezza.

Altri strumenti

- Python è un linguaggio di programmazione per computer generico.

Deposito di codice

Il codice per questo modello è disponibile nell'archivio di certificati CA GitHub Detect Amazon RDS con certificati CA in scadenza.

Best practice

Ti consigliamo di attenerti alle best practice riportate nelle seguenti risorse:

- Best practice per le unità organizzative con AWS Organizations (blog AWS Cloud Operations & Migrations)
- Guida per la creazione di una base iniziale utilizzando AWS Control Tower on AWS (AWS Solutions Library)
- Linee guida per la creazione e la modifica AWS Control Tower delle risorse (AWS Control Tower documentazione)
- Considerazioni sull'implementazione del cFCT (documentazione)AWS Control Tower

Epiche

Esamina la soluzione e il codice

Operazione	Description	Competenze richieste
Determina la tua strategia di implementazione.	Esamina la soluzione e il codice per determinare come implementarli nel tuo AWS ambiente. Determina se intendi eseguire la distribuz ione in un unico account o in un' AWS organizzazione.	Proprietario dell'app, General AWS
Clonare il repository.	Immetti il seguente comando per clonare l'archivio di certificati CA Detect Amazon RDS con certificati CA in scadenza. `git clone https://github.com/aws-samples/config-rds-ca-expiry.git`	Sviluppatore di app, proprietario dell'app
Convalida la versione di Python.	1. Naviga nella directory di primo livello nel repository clonato.	Sviluppatore di app, proprietario dell'app

Operazione	Description	Competenze richieste
	`cd config-rds-ca-expiry` 2. config-rds-ca-expiryApri.yaml. 3. Nella `CertExpirationCheckLambdaFunction` risorsa, verifica che la versione di Python sia compatibile con il tuo obiettivo. Regioni AWS Per impostazione predefinita, questa funzione utilizza Python 3.12. Per ulteriori informazioni, consulta AWS Lambda aggiunge il supporto per Python 3.12. Se necessario, aggiorna la versione di Python. 4. Salva e chiudi config-rds-ca-expiry.yaml.	

Implementazione della soluzione

Operazione	Description	Competenze richieste
Implementa il CloudFormation modello.	Implementa il CloudFormation modello nel tuo AWS ambiente. Esegui una delle seguenti operazioni: • Se stai eseguendo la distribuzione su un singolo dispositivo Account AWS,	Sviluppatore di app, amministratore AWS, General AWS

Operazione	Description	Competenze richieste
	segui le istruzioni riportate in Creazione di uno stack. • Se stai eseguendo la distribuzione in un'organizzazione che non è gestita da AWS Control Tower, segui le istruzioni in Creare un set di stack. • Se esegui la distribuzione in un'organizzazione gestita da AWS Control Tower, consulta le istruzioni nella sezione Crea le tue personalizzazioni.	
Verifica la distribuzione.	Nella CloudFormation console, verifica che lo stack o il set di stack siano stati distribuiti correttamente.	Amministratore AWS, proprietario dell'app

Esamina i risultati

Operazione	Description	Competenze richieste
Visualizza i risultati delle AWS Config regole.	In Security Hub CSPM, procedi come segue per visualizzare un elenco di risultati individuali: 1. Apri la console Security Hub CSPM. 2. Nel riquadro di navigazione, seleziona Esiti.	Amministratore AWS, amministratore di sistema AWS, amministratore cloud

Operazione	Description	Competenze richieste
	3. Nella casella Aggiungi filtri, aggiungi i seguenti filtri: • Lo stato di conformità è `FAILED` • Il titolo è `rds-has-expiring-ca` 4. Scegli Applica. In Security Hub CSPM, procedi come segue per visualizzare un elenco di risultati totali raggruppati per: Account AWS 1. Apri la console Security Hub CSPM. 2. Nel riquadro di navigazione, seleziona Informazioni dettagliate. 3. Scegliere Create insight (Crea informazione dettagliata). 4. Per selezionare l'attributo di raggruppamento per l'informazione dettagliata: a. Scegli la casella di ricerca per visualizzare le opzioni di filtro. b. Scegliere Group by (Raggruppa per). c. Seleziona AwsAccountId. d. Scegli Applica.	

Operazione	Description	Competenze richieste
	5. Nella casella Aggiungi filtri, aggiungi i seguenti filtri: • Il titolo è `rds-has-expiring-ca` • Lo stato di conformità è `FAILED` 6. Scegliere Create insight (Crea informazione dettagliata). 7. Inserisci un nome per Insight, quindi scegli Create insight. Inoltre AWS Config, per visualizzare un elenco di risultati, segui le istruzioni in Visualizzazione delle informazioni sulla conformità e dei risultati della valutazione nella AWS Config documentazione.	

risoluzione dei problemi

Problema	Soluzione
CloudFormation la creazione o l'eliminazione dello stack set fallisce	Quando AWS Control Tower viene distribuito, impone i guardrail necessari e assume il controllo degli aggregatori e delle regole. AWS Config Ciò include la prevenzione di eventuali modifiche dirette. CloudFormation Per distribuire o rimuovere correttamente questo

Problema	Soluzione
	CloudFormation modello, incluse tutte le risorse associate, è necessario utilizzare cFCT.
cFct non riesce a eliminare il modello CloudFormation	Se il CloudFormation modello persiste anche dopo aver apportato le modifiche necessarie nel file manifesto e aver rimosso i file modello, verificate che il file manifesto contenga il `enable_stack_set_deletion` parametro e che il valore sia impostato su. `false` Per ulteriori informazioni, vedete Eliminare uno stack set nella documentazione cFct.

Risorse correlate

- Utilizzo SSL/TLS per crittografare una connessione a un'istanza o un cluster DB (documentazione Amazon RDS)
- AWS Config Regole personalizzate (documentazione)AWS Config

Genera dinamicamente una policy IAM con IAM Access Analyzer utilizzando Step Functions

Thomas Scott, Koen van Blijderveen, Adil El Kanabi e Rafal Pawlaszek, Amazon Web Services

Riepilogo

Least-Privilege è la migliore pratica di sicurezza per concedere le autorizzazioni minime necessarie per eseguire un'attività. Implementare l'accesso con privilegi minimi in un account Amazon Web Services (AWS) già attivo può essere difficile perché non si desidera impedire involontariamente agli utenti di svolgere le proprie mansioni lavorative modificando le loro autorizzazioni. Prima di poter implementare le modifiche alle policy AWS Identity and Access Management (IAM), devi comprendere le azioni e le risorse eseguite dagli utenti dell'account.

Questo modello è progettato per aiutarti ad applicare il principio dell'accesso con privilegi minimi, senza bloccare o rallentare la produttività del team. Descrive come utilizzare IAM Access Analyzer e AWS Step Functions generare dinamicamente una policy up-to-date IAM per il tuo ruolo, in base alle azioni attualmente eseguite nell'account. La nuova policy è progettata per consentire l'attività corrente ma rimuovere eventuali privilegi elevati e non necessari. È possibile personalizzare la policy generata definendo regole di autorizzazione e rifiuto e la soluzione integra le regole personalizzate.

Questo modello include opzioni per implementare la soluzione con AWS Cloud Development Kit (AWS CDK) o HashiCorp CDK for Terraform (CDKTF). È quindi possibile associare la nuova policy al ruolo utilizzando una pipeline di integrazione e distribuzione continue (CI/CD). Se disponi di un'architettura multi-account, puoi implementare questa soluzione in qualsiasi account in cui desideri generare policy IAM aggiornate per i ruoli, aumentando la sicurezza dell'intero ambiente. Cloud AWS

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS con un AWS CloudTrail trail abilitato.
- Autorizzazioni IAM per quanto segue:
  - Crea e distribuisci flussi di lavoro Step Functions. Per ulteriori informazioni, consulta Azioni, risorse e chiavi di condizione per AWS Step Functions (documentazione Step Functions).
  - Creare AWS Lambda funzioni. Per ulteriori informazioni, consulta Ruolo di esecuzione e autorizzazioni utente (documentazione Lambda).

  - Creare ruoli IAM. Per ulteriori informazioni, consulta Creazione di un ruolo per delegare le autorizzazioni a un utente IAM (documentazione IAM).
- npm installato. Per ulteriori informazioni, vedere Download e installazione di Node.js e npm (documentazione di npm).
- Se si distribuisce questa soluzione con AWS CDK (Opzione 1):
  - AWS CDK Toolkit, installato e configurato. Per ulteriori informazioni, consulta Installare il AWS CDK (AWS CDK documentazione).
- Se state distribuendo questa soluzione con CDKTF (opzione 2):
  - CDKTF, installato e configurato. Per ulteriori informazioni, consulta Install CDK for Terraform (documentazione CDKTF).
  - Terraform, installato e configurato. Per ulteriori informazioni, consulta Get Started (documentazione Terraform).
- AWS Command Line Interface (AWS CLI) installato e configurato localmente per il tuo Account AWS. Per ulteriori informazioni, vedere Installazione o aggiornamento della versione più recente di AWS CLI (AWS CLI documentazione).

Limitazioni

- Questo modello non applica la nuova policy IAM al ruolo. Al termine di questa soluzione, la nuova policy IAM viene archiviata in un AWS CodeCommit repository. Puoi utilizzare una CI/CD pipeline per applicare le policy ai ruoli del tuo account.

Architecture

Architettura Target

1. Una regola di EventBridge eventi Amazon pianificata regolarmente avvia un flusso di lavoro Step Functions. Questo programma di rigenerazione viene definito come parte della configurazione di questa soluzione.
2. Nel flusso di lavoro Step Functions, una funzione Lambda genera gli intervalli di date da utilizzare per analizzare l'attività dell'account nei registri. CloudTrail
3. La fase successiva del flusso di lavoro richiama l'API IAM Access Analyzer per iniziare a generare la policy.

4. Utilizzando l'Amazon Resource Name (ARN) del ruolo specificato durante la configurazione, IAM Access Analyzer analizza CloudTrail i log per individuare le attività entro la data specificata. In base all'attività, IAM Access Analyzer genera una policy IAM che consente solo le azioni e i servizi utilizzati dal ruolo durante l'intervallo di date specificato. Una volta completato questo passaggio, questo passaggio genera un ID del lavoro.
5. La fase successiva del flusso di lavoro verifica l'ID del lavoro ogni 30 secondi. Quando viene rilevato l'ID del lavoro, questo passaggio utilizza l'ID del lavoro per chiamare l'API IAM Access Analyzer e recuperare la nuova policy IAM. IAM Access Analyzer restituisce la policy come file JSON.
6. La fase successiva del flusso di lavoro inserisce il file /policy.json <IAM role name>in un bucket Amazon Simple Storage Service (Amazon S3). Definisci questo bucket S3 come parte della configurazione di questa soluzione.
7. Una notifica di evento Amazon S3 avvia una funzione Lambda.
8. La funzione Lambda recupera la policy dal bucket S3, integra le regole personalizzate definite nei file allow.json e deny.json, quindi invia la policy aggiornata a. CodeCommit Il CodeCommit repository, il ramo e il percorso della cartella vengono definiti durante la configurazione di questa soluzione.

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che consente di definire e fornire Cloud AWS l'infrastruttura nel codice.
- AWS CDK Toolkit è un kit di sviluppo cloud a riga di comando che ti aiuta a interagire con la tua AWS Cloud Development Kit (AWS CDK) app.
- AWS CloudTrailti aiuta a controllare la governance, la conformità e il rischio operativo del tuo Account AWS.
- AWS CodeCommitè un servizio di controllo delle versioni che consente di archiviare e gestire in modo privato gli archivi Git, senza dover gestire il proprio sistema di controllo del codice sorgente.
- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle. Questo modello utilizza IAM Access Analyzer, una funzionalità di IAM, per analizzare CloudTrail i log per identificare le azioni e

i servizi che sono stati utilizzati da un'entità IAM (utente o ruolo) e quindi generare una policy IAM basata su tale attività.

- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Step Functionsè un servizio di orchestrazione senza server che consente di combinare AWS Lambda funzioni e altro Servizi AWS per creare applicazioni aziendali critiche. In questo modello, utilizzi le integrazioni dei servizi AWS SDK in Step Functions per richiamare le azioni dell'API di servizio dal tuo flusso di lavoro.

Altri strumenti

- CDK for Terraform (CDKTF) ti aiuta a definire l'infrastruttura come codice (IaC) utilizzando linguaggi di programmazione comuni, come Python e Typescript.
- Lerna è un sistema di compilazione per la gestione e la pubblicazione di più pacchetti o pacchetti dallo stesso repository. JavaScript TypeScript
- Node.js è un ambiente di JavaScript runtime basato sugli eventi progettato per la creazione di applicazioni di rete scalabili.
- npm è un registro software che viene eseguito in un ambiente Node.js e viene utilizzato per condividere o prendere in prestito pacchetti e gestire la distribuzione di pacchetti privati.

Archivio di codice

Il codice di questo modello è disponibile nel repository GitHub Automated IAM Access Analyzer Role Policy Generator.

Epiche

Preparati per l'implementazione

Operazione	Description	Competenze richieste
Clona il repository.	Il comando seguente clona il repository Automated IAM	Sviluppatore di app

Operazione	Description	Competenze richieste
	Access Analyzer Role Policy Generator (). GitHub `git clone https://github.com/aws-samples/automated-iam-access-analyzer.git`	
Installa Lerna.	Il comando seguente installa Lerna. `npm i -g lerna`	Sviluppatore di app

Imposta le dipendenze.	Il comando seguente installa le dipendenze per il repository.
	`cd automated-iam-access-analyzer/`
	`npm install && npm run bootstrap`
		Sviluppatore di app

Compila il codice.	Il comando seguente verifica, crea e prepara i pacchetti zip delle funzioni Lambda.
	`npm run test:code`
	`npm run build:code`
	`npm run pack:code`
		Sviluppatore di app

Costruisci i costrutti.	Il comando seguente crea le applicazioni di sintesi dell'infrastruttura, sia per CDKTF che per CDKTF. AWS CDK `npm run build:infra`	

Operazione	Description	Competenze richieste
Configura eventuali autorizza zioni personalizzate.	Nella cartella repo del repository clonato, modifica i file allow.json e deny.json per definire eventuali autorizza zioni personalizzate per il ruolo. Se i file allow.json e deny.json contengono la stessa autorizzazione, viene applicata l'autorizzazione di negazione.	Amministratore AWS, sviluppatore di app

Opzione 1: distribuisci la soluzione utilizzando AWS CDK

Operazione	Description	Competenze richieste
Distribuisci lo AWS CDK stack.	Il comando seguente distribui sce l'infrastruttura tramite. AWS CloudFormation Definire i seguenti parametri: • `<NAME_OF_ROLE>` — L'ARN del ruolo IAM per il quale si sta creando una nuova policy. • `<TRAIL_ARN>` — L'ARN del CloudTrail percorso in cui è memorizzata l'attività del ruolo. • `<CRON_EXPRESSION_TO_RUN_SOLUTION>` — L'espressione Cron che definisce il programma di rigenerazione della policy. Il flusso di lavoro Step	Sviluppatore di app

Operazione	Description	Competenze richieste
	Functions viene eseguito secondo questa pianificazione. • <TRAIL_LOOKBACK> — Il periodo, in giorni, necessario per esaminare il passato nella valutazione delle autorizzazioni dei ruoli. cd infra/cdk cdk deploy —-parameters roleArn=<NAME_OF_ROLE> \ —-parameters trailArn=<TRAIL_ARN> \ --parameters schedule=<CRON_EXPRESSION_TO_RUN_SOLUTION> \ [--parameters trailLookBack=<TRAIL_LOOKBACK>] Note Le parentesi quadre indicano parametri opzionali.	

Operazione	Description	Competenze richieste
(Facoltativo) Attendi la nuova politica.	Se l'itinerario non contiene una quantità ragionevole di attività storiche per il ruolo, attendi di avere la certezza che ci sia abbastanza attività registrata da consentire a IAM Access Analyzer di generare una policy accurata. Se il ruolo è attivo nell'acco unt da un periodo di tempo sufficiente, questo periodo di attesa potrebbe non essere necessario.	Amministratore AWS
Esamina manualmente la policy generata.	Nel tuo CodeCommit repository, esamina il file.json <ROLE_ARN>generato per confermare che le autorizza zioni di autorizzazione e negazione siano appropriate per il ruolo.	Amministratore AWS

Opzione 2: implementa la soluzione utilizzando CDKTF

Operazione	Description	Competenze richieste
Sintetizza il modello Terraform .	Il comando seguente sintetizz a il modello Terraform. `lerna exec cdktf synth --scope @aiaa/tfm`	Sviluppatore di app
Distribuisci il modello Terraform.	Il comando seguente accede alla directory che contiene	Sviluppatore di app

Operazione	Description	Competenze richieste
	l'infrastruttura definita da CDKTF. `cd infra/cdktf` Il comando seguente distribui sce l'infrastruttura nella destinazione. Account AWS Definire i seguenti parametri: • `<account_ID>` — L'ID dell'account di destinazione. • `<region>`- L'obiettivo Regione AWS. • `<selected_role_ARN >` — L'ARN del ruolo IAM per il quale si sta creando una nuova policy. • `<trail_ARN>` — L'ARN del CloudTrail percorso in cui è memorizzata l'attività del ruolo. • `<schedule_expressi on>` — L'espressione Cron che definisce il programma di rigenerazione della policy. Il flusso di lavoro Step Functions viene eseguito secondo questa pianifica zione. • `<trail_look_back>` — Il periodo, in giorni, necessari o per esaminare il passato	

Operazione	Description	Competenze richieste
	nella valutazione delle autorizzazioni dei ruoli. `TF_VAR_accountId=<account_ID> \ TF_VAR_region=<region> \ TF_VAR_roleArns=<selected_role_ARN> \ TF_VAR_trailArn=<trail_ARN> \ TF_VAR_schedule=<schedule_expression> \ [ TF_VAR_trailLookBack=<trail_look_back> ] \ cdktf deploy` **Note** Le parentesi quadre indicano parametri opzionali.	

Operazione	Description	Competenze richieste
(Facoltativo) Attendi la nuova politica.	Se l'itinerario non contiene una quantità ragionevole di attività storiche per il ruolo, attendi di avere la certezza che ci sia abbastanza attività registrata da consentire a IAM Access Analyzer di generare una policy accurata. Se il ruolo è attivo nell'acco unt da un periodo di tempo sufficiente, questo periodo di attesa potrebbe non essere necessario.	Amministratore AWS
Esamina manualmente la policy generata.	Nel tuo CodeCommit repository, esamina il file.json <ROLE_ARN>generato per confermare che le autorizza zioni di autorizzazione e negazione siano appropriate per il ruolo.	Amministratore AWS

Risorse correlate

AWS resources

- Endpoint e quote IAM Access Analyzer
- Configurazione di AWS CLI
- Iniziare con AWS CDK
- Autorizzazioni con privilegi minimi

Altre risorse

- CDK per Terraform (sito web Terraform)

Abilita Amazon in GuardDuty modo condizionale utilizzando modelli AWS CloudFormation

Ram Kandaswamy, Amazon Web Services

Riepilogo

AWS CloudFormation, uno strumento di infrastruttura come codice (IaC), ti aiuta a gestire AWS le risorse tramite implementazioni basate su modelli. CloudFormation viene in genere utilizzato per gestire le risorse. AWS Usarlo per abilitare Servizi AWS, come Amazon GuardDuty, può presentare sfide uniche. GuardDuty è un servizio di rilevamento delle minacce che monitora continuamente Account AWS le attività dannose e i comportamenti non autorizzati. A differenza delle risorse tipiche che possono essere create più volte, GuardDuty è un servizio che deve essere abilitato una volta per account e. Regione AWS CloudFormation Le condizioni tradizionali supportano solo confronti di valori statici, il che rende difficile verificare lo stato attuale di servizi come GuardDuty. Se tenti di GuardDuty abilitarlo CloudFormation in un account in cui è già attivo, la distribuzione dello stack fallisce. Ciò può creare sfide operative per i DevOps team che gestiscono ambienti con più account.

Questo modello introduce una soluzione a questa sfida. Utilizza risorse CloudFormation personalizzate supportate da AWS Lambdafunzioni per eseguire controlli di stato dinamici. La logica condizionale si attiva GuardDuty solo se non è già abilitata. Utilizza gli output dello stack per registrare lo GuardDuty stato per riferimenti futuri.

Seguendo questo schema, è possibile automatizzare le GuardDuty implementazioni nell' AWS infrastruttura mantenendo al contempo operazioni di stack pulite e prevedibili. CloudFormation Questo approccio è particolarmente utile per le organizzazioni che sono:

- Gestione multipla Account AWS tramite IaC
- Implementazione di servizi di sicurezza su larga scala
- Richiede implementazioni di infrastrutture idempotenti
- Automatizzazione delle implementazioni dei servizi di sicurezza

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS

- Un ruolo AWS Identity and Access Management (IAM) che dispone delle autorizzazioni per creare, aggiornare ed eliminare gli stack CloudFormation
- AWS Command Line Interface (AWS CLI), installato e configurato

Limitazioni

Se GuardDuty è stato disabilitato manualmente per un Account AWS o Regione AWS, questo pattern non è abilitato GuardDuty per quell'account o regione di destinazione.

Architecture

Stack tecnologico Target

Il modello utilizza CloudFormation l'infrastruttura come codice (IaC). Si utilizza una risorsa CloudFormation personalizzata supportata da una funzione Lambda per ottenere la funzionalità di abilitazione dinamica dei servizi.

Architettura Target

Il seguente diagramma di architettura di alto livello mostra il processo di abilitazione GuardDuty mediante l'implementazione di un modello: CloudFormation

1. Implementate un CloudFormation modello per creare uno CloudFormation stack.
2. Lo stack crea un ruolo IAM e una funzione Lambda.
3. La funzione Lambda assume il ruolo IAM.
4. Se non GuardDuty è già abilitato sulla destinazione Account AWS, la funzione Lambda lo abilita.

Automazione e scalabilità

È possibile utilizzare la AWS CloudFormation StackSet funzionalità per estendere questa soluzione a più Account AWS e Regioni AWS. Per ulteriori informazioni, consulta Lavorare con AWS CloudFormation StackSets nella CloudFormation documentazione.

Tools (Strumenti)

- AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire Servizi AWS tramite comandi nella shell della riga di comando.

- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita in tutte le regioni. Account AWS
- Amazon GuardDuty è un servizio di monitoraggio continuo della sicurezza che analizza ed elabora i log per identificare attività impreviste e potenzialmente non autorizzate nel tuo ambiente. AWS
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue AWS risorse controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.

Epiche

Crea il CloudFormation modello e distribuisci lo stack

Operazione	Description	Competenze richieste
Memorizza il codice in Amazon S3.	1. Copia il codice Python nella sezione Informazioni aggiuntive di questo modello. 2. Incolla il codice in un editor di testo. 3. Salva il file con nome `index.py`. 4. Caricare il file in un bucket Amazon Simple Storage Service (Amazon S3). Per istruzioni, consulta Caricamento di oggetti nella documentazione di Amazon S3.	AWS DevOps
Crea il CloudFormation modello.	1. Apri la CloudFormation console.	AWS DevOps

Operazione	Description	Competenze richieste
	2. Nel riquadro di navigazione a sinistra, scegli Infrastructure Composer. 3. Se non vedi una tela bianca, crea un nuovo progetto. 4. Trascina e rilascia una AWS Lambda funzione sull'area di disegno. 5. Nella vista modello, verifica che siano presenti una funzione Lambda e un gruppo di log. 6. Modifica `Runtime` la funzione Lambda all'ultima versione di Python. 7. Verifica che la `Handler` proprietà abbia il valore di `index.lambda_handler`. 8. Configura la `CodeUri` proprietà come posizione Amazon S3 in cui hai caricato il codice Python. Un valore di esempio è `s3://amzn-s3-demo-bucket/key-name` . 9. Aggiungi una `Policies` proprietà per la risorsa. Segui le migliori pratiche di sicurezza per fornire autorizzazioni di accesso CloudFormation e	

Operazione	Description	Competenze richieste

	GuardDuty azioni con privilegi minimi. Ad esempio, è possibile utilizzare la politica `AWSLambdaExecute` gestita per i log e un modulo personalizzato. `iam:CreateServiceLinkedRole` GuardDuty
	10. Aggiungi una definizione di risorsa personalizzata allo snippet accedendo alla visualizzazione del modello.
	`CheckResourceExist:`
	`  Type: 'Custom::LambdaCustomResource'`
	`  Properties:`
	`  ServiceToken: !GetAtt Function.Arn`
	11. Salva il modello come. `sample.yaml`	

Operazione	Description	Competenze richieste

Crea lo CloudFormation stack.	1. Nel AWS CLI, inserisci il seguente comando. Questo crea un nuovo CloudFormation stack utilizzando il `sample.yaml` file. Per ulteriori informazioni, consulta Creazione di uno stack nella CloudFormation documentazione.
	`aws cloudformation create-stack \ --stack-name guardduty-cf-stack \ --template-body file://sample.yaml`
	2. Conferma che il seguente valore sia visualizzato in AWS CLI, a indicare che lo stack è stato creato con successo. La quantità di tempo necessaria per creare lo stack può variare.
	`"StackStatus": "CREATE_COMPLETE",`
		AWS DevOps

Convalida che GuardDuty sia abilitato per. Account AWS	1. Accedi a Console di gestione AWS e apri la GuardDuty console. 2. Verifica che il GuardDuty servizio sia abilitato.	Amministratore cloud, amministratore AWS

Operazione	Description	Competenze richieste
Configura account o regioni aggiuntivi.	Se necessario per il tuo caso d'uso, utilizza la CloudForm ation StackSet funzionalità per estendere questa soluzione a più Account AWS e Regioni AWS. Per ulteriori informazi oni, consulta Lavorare con AWS CloudFormation StackSets nella CloudForm ation documentazione.	Amministratore cloud, amministratore AWS

Risorse correlate

Riferimenti

- AWS CloudFormation documentazione
- AWS Lambda riferimento al tipo di risorsa
- CloudFormation tipo di risorsa: AWS: :IAM: :Role
- CloudFormation tipo di risorsa:: AWS: GuardDuty :Detector
- Quattro modi per recuperare qualsiasi proprietà del AWS servizio utilizzando AWS CloudFormation (post sul blog)

Tutorial e video

- Semplifica la gestione dell'infrastruttura utilizzando AWS CloudFormation(tutorial)
- Usa Amazon GuardDuty e AWS Security Hub CSPM per proteggere più account (AWS re:Invent 2020)
- Le migliori pratiche per la creazione (re:Invent 2019 AWS CloudFormation)AWS
- Rilevamento delle minacce su AWS: un'introduzione ad Amazon GuardDuty (AWS re:InForce 2019)

Informazioni aggiuntive

Codice Python

```
import boto3
import os
import json
from botocore.exceptions import ClientError
import cfnresponse
guardduty=boto3.client('guardduty')
cfn=boto3.client('cloudformation')
def lambda_handler(event, context):
    print('Event: ', event)
    if 'RequestType' in event:
      if event['RequestType'] in ["Create","Update"]:
          enabled=False
          try:
            response=guardduty.list_detectors()
            if "DetectorIds" in response and len(response["DetectorIds"])>0:
              enabled="AlreadyEnabled"
            elif "DetectorIds" in response and len(response["DetectorIds"])==0:
              cfn_response=cfn.create_stack(
                StackName='guardduty-cfn-stack',
                TemplateBody='{ "AWSTemplateFormatVersion": "2010-09-09",
    "Description": "Guard duty creation template",    "Resources":
 { "IRWorkshopGuardDutyDetector": {  "Type": "AWS::GuardDuty::Detector",
 "Properties": {   "Enable": true  }   } } }'
                )
              enabled="True"
          except Exception as e:
              print("Exception: ",e)
          responseData = {}
          responseData['status'] = enabled
          cfnresponse.send(event, context, cfnresponse.SUCCESS, responseData,
 "CustomResourcePhysicalID" )
      elif event['RequestType'] == "Delete":
          cfn_response=cfn.delete_stack(
                  StackName='guardduty-cfn-stack')
          cfnresponse.send(event, context, cfnresponse.SUCCESS, {})
```

Abilita la crittografia trasparente dei dati in Amazon RDS for SQL Server

Ranga Cherukuri, Amazon Web Services

Riepilogo

Questo modello descrive come implementare la crittografia trasparente dei dati (TDE) in Amazon Relational Database Service (Amazon RDS) per SQL Server per crittografare i dati inattivi.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un'istanza DB di Amazon RDS per SQL Server

Versioni del prodotto

Amazon RDS attualmente supporta TDE per le seguenti versioni ed edizioni di SQL Server:

- SQL Server 2016 Enterprise Edition
- SQL Server 2017 Enterprise Edition
- SQL Server 2019 Standard ed Enterprise Edition
- SQL Server 2022 Standard ed Enterprise Edition

Per le informazioni più recenti sulle versioni e le edizioni supportate, consulta Support for Transparent Data Encryption in SQL Server nella documentazione di Amazon RDS.

Architecture

Stack tecnologico

- Amazon RDS per SQL Server

Architettura

Tools (Strumenti)

- Microsoft SQL Server Management Studio (SSMS) è un ambiente integrato per la gestione di un'infrastruttura SQL Server. Fornisce un'interfaccia utente e un gruppo di strumenti con editor di script avanzati che interagiscono con SQL Server.

Epiche

Crea un gruppo di opzioni nella console Amazon RDS

Operazione	Description	Competenze richieste
Apri la console Amazon RDS.	Accedi alla Console di gestione AWS e apri la console Amazon RDS.	Sviluppatore, DBA
Crea un gruppo di opzioni.	Nel riquadro di navigazione, scegli Gruppi di opzioni, Crea gruppo. Scegli sqlserver-ee come motore di database, quindi seleziona la versione del motore.	Sviluppatore, DBA
Aggiungete l'opzione TRANSPARENT_DATA_ENCRYPTION.	Modificate il gruppo di opzioni creato e aggiungete l'opzione chiamata. `TRANSPARENT_DATA_ENCRYPTION`	Sviluppatore, DBA

Associare il gruppo di opzioni a questa istanza database

Operazione	Description	Competenze richieste
Scegli l'istanza DB.	Nella console Amazon RDS, nel pannello di navigazione, scegli Database, quindi scegli	Sviluppatore, DBA

Operazione	Description	Competenze richieste
	l'istanza DB che desideri associare al gruppo di opzioni.	
Associa l'istanza DB al gruppo di opzioni.	Scegliete Modifica, quindi utilizzate l'impostazione del gruppo di opzioni per associare l'istanza DB di SQL Server al gruppo di opzioni creato in precedenza.	Sviluppatore, DBA
Applica le modifiche.	Applica le modifiche immediatamente o durante la finestra di manutenzione successiva, come desiderato.	Sviluppatore, DBA

Ottieni il nome del certificato.	Ottieni il nome del certifica to predefinito utilizzando la seguente query.
	`USE [master]`
	`GO`
	`SELECT name FROM sys.certificates WHERE name LIKE 'RDSTDECertificate%'`
	`GO`
		Sviluppatore, DBA

Crea la chiave di crittografia del database

Operazione	Description	Competenze richieste
Connect all'istanza DB di Amazon RDS for SQL Server tramite SSMS.	Per istruzioni, consulta Using SSMS nella documentazione Microsoft.	Sviluppatore, DBA

Operazione	Description	Competenze richieste

Crea la chiave di crittografia del database utilizzando il certificato predefinito.	Crea una chiave di crittografia del database utilizzando il nome di certificato predefinito che hai ricevuto in precedenza. Usa la seguente query T-SQL per creare una chiave di crittografia del database. È possibile specificare l'algoritmo AES_256 anziché AES_128.
	`USE [Databasename]`
	`GO`
	`CREATE DATABASE`
	`ENCRYPTION KEY`
	`WITH ALGORITHM = AES_128`
	`ENCRYPTION BY SERVER`
	`CERTIFICATE [certificatename]`
	`GO`
		Sviluppatore, DBA

Abilita la crittografia sul database.	Usa la seguente query T-SQL per abilitare la crittografia del database.
	`ALTER DATABASE [Database Name]`
	`SET ENCRYPTION ON`
	`GO`
		Sviluppatore, DBA

Operazione	Description	Competenze richieste
Controlla lo stato della crittografia.	Usa la seguente query T-SQL per verificare lo stato della crittografia. `SELECT DB_NAME(database_id) AS DatabaseName, encryption_state, percent_complete FROM sys.dm_database_encryption_keys`	Sviluppatore, DBA

Risorse correlate

- Support per la crittografia trasparente dei dati in SQL Server (documentazione Amazon RDS)
- Utilizzo dei gruppi di opzioni (documentazione Amazon RDS)
- Modifica di un'istanza database Amazon RDS (documentazione Amazon RDS)
- Crittografia trasparente dei dati per SQL Server (documentazione Microsoft)
- Utilizzo di SSMS (documentazione Microsoft)

Monitora e correggi l'eliminazione pianificata delle chiavi AWS KMS

Mikesh Khanal e Ramya Pulipaka, Amazon Web Services

Riepilogo

Sul cloud Amazon Web Services (AWS), l'eliminazione di una chiave AWS Key Management Services (AWS KMS) può causare la perdita di dati. L'eliminazione rimuove il materiale chiave e tutti i metadati associati alla chiave AWS KMS ed è irreversibile. Dopo l'eliminazione di una chiave AWS KMS, non è più possibile decrittografare i dati crittografati con quella chiave AWS KMS, in modo che i dati non possano essere recuperati.

Questo modello imposta il monitoraggio, con notifiche quando un'applicazione o un utente pianifica l'eliminazione di una chiave AWS KMS. Se ricevi una notifica, potresti voler annullare l'eliminazione della chiave AWS KMS e riconsiderare la tua decisione di eliminarla. Il modello utilizza il runbook di automazione di AWS Systems Manager AWSConfigRemediation-CancelKeyDeletionper facilitare l'annullamento dell'eliminazione di una chiave AWS KMS.

> **Note**
>
> Il CloudFormation modello del pattern deve essere distribuito in tutte le regioni AWS in cui desideri monitorare l'eliminazione delle chiavi AWS KMS.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Comprensione dei seguenti servizi AWS:
  - Amazon EventBridge
  - AWS KMS
  - Amazon Simple Notification Service (Amazon SNS)
  - AWS Systems Manager

Limitazioni

- Qualsiasi personalizzazione della soluzione richiede la conoscenza dei CloudFormation modelli AWS e dei servizi AWS utilizzati in questo modello.
- Attualmente, questa soluzione utilizza il bus di eventi predefinito e può essere personalizzata in base ai requisiti. Per ulteriori informazioni sul bus di eventi personalizzato, consulta la documentazione AWS.

Architecture

Stack tecnologico Target

- Amazon EventBridge
- AWS KMS
- Amazon SNS
- AWS Systems Manager
- Automazione utilizzando quanto segue:
  - AWS Command Line Interface (AWS CLI) o SDK AWS
  - CloudFormation Stack AWS

Architettura Target

1. L'eliminazione di una chiave AWS KMS è pianificata.
2. L'evento di eliminazione pianificata viene valutato in base a una regola. EventBridge
3. La EventBridge regola riguarda l'argomento Amazon SNS.
4. La EventBridge regola avvia l'automazione e i runbook di Systems Manager.
5. I runbook annullano l'eliminazione.

Automazione e scalabilità

Lo CloudFormation stack implementa tutte le risorse e i servizi necessari per il funzionamento di questa soluzione. Il pattern può essere eseguito indipendentemente in un singolo account o eseguito utilizzando AWS CloudFormation StackSets per più account indipendenti o un'organizzazione.

```
aws cloudformation create-stack --stack-name  <stack-name>\
```

```
    --template-body file://<Full-Path-of-file> \
    --parameters ParameterKey=,ParameterValue= \
    --capabilities CAPABILITY_NAMED_IAM
```

Tools (Strumenti)

Strumenti

- AWS CloudFormation: AWS CloudFormation è un servizio che ti aiuta a modellare e configurare le tue risorse Amazon Web Services in modo da poter dedicare meno tempo alla gestione di tali risorse e più tempo a concentrarti sulle applicazioni eseguite su AWS. Puoi utilizzare un CloudFormation modello per creare stack in un account AWS in una regione AWS. Il modello descrive tutte le risorse AWS che desideri, effettua il CloudFormation provisioning e configura tali risorse per te.
- AWS CLI — L'AWS Command Line Interface (AWS CLI) è uno strumento open source che puoi usare per interagire con i servizi AWS utilizzando i comandi nella shell della riga di comando.
- Amazon EventBridge: Amazon EventBridge è un servizio di bus eventi senza server che collega le tue applicazioni con dati provenienti da una varietà di fonti. EventBridge fornisce un flusso di dati in tempo reale dalle tue applicazioni e dai servizi AWS e indirizza tali dati verso obiettivi come AWS Lambda. EventBridge semplifica il processo di creazione di architetture basate sugli eventi.
- AWS KMS — AWS Key Management Service (AWS KMS) è un servizio gestito per la creazione e il controllo delle chiavi AWS KMS, le chiavi di crittografia utilizzate per crittografare i dati.
- AWS SDKs: gli strumenti SDKs AWS consentono di sviluppare e gestire applicazioni su AWS nel linguaggio di programmazione che preferisci.
- Amazon SNS — Amazon Simple Notification Service (Amazon SNS) è un servizio gestito che fornisce il recapito dei messaggi dagli editori agli abbonati (noti anche come produttori e consumatori). Gli editori comunicano in modo asincrono con gli abbonati creando e inviando messaggi a un argomento, che rappresenta un punto di accesso logico e un canale di comunicazione.
- AWS Systems Manager — AWS Systems Manager è un servizio AWS che puoi usare per visualizzare e controllare la tua infrastruttura su AWS. Utilizzando la console Systems Manager, puoi automatizzare le attività operative tra le tue risorse AWS. Systems Manager consente di mantenere la sicurezza e la conformità eseguendo la scansione delle Istanze gestite e segnalando eventuali violazioni dei criteri rilevate (o intraprendendo azioni correttive in merito).

Codice

- Il `alerting_ct_logs.yaml` CloudFormation modello per il progetto è allegato.

Epiche

Prepara l'account AWS

Operazione	Description	Competenze richieste
Installa e configura AWS CLI.	Installa la versione 2 dell'interfaccia a riga di comando di AWS. Quindi configura le impostazioni delle credenziali di sicurezza per un'identità, il formato di output predefinito e la regione AWS predefinita che AWS CLI utilizza per interagire con AWS. L'identità deve disporre delle autorizzazioni necessarie per eseguire le attività.	Sviluppatore, ingegnere della sicurezza

Implementa il modello AWS CloudFormation

Operazione	Description	Competenze richieste
Scarica il CloudFormation modello.	Scaricate l'allegato in un percorso locale sul computer ed estraete il file `alerting_ct_logs.yaml` modello.	Sviluppatore, ingegnere della sicurezza
Implementa il modello.	Nella finestra del terminale in cui è stato configurato il profilo dell'account AWS, esegui il comando seguente.	Sviluppatore, ingegnere della sicurezza

Operazione	Description	Competenze richieste
	`aws cloudformation create-stack --stack-name <stack_name> \ --capabilities <Value> \ --template-body file://<Full_Path> \ --parameters ParameterKey=DestinationEmailAddress,ParameterValue=<Value> \ ParameterKey=SNSTopicName,ParameterValue=<Value> \ ParameterKey=EnableRemediation ,ParameterValue=<Value> \ ParameterKey=AutomationAssumeRole, ParameterValue=<Value>` Nel passaggio successivo, inserisci i valori per i parametri del modello.	

Operazione	Description	Competenze richieste
Completa i parametri del modello.	Immettete i valori richiesti per i parametri. • `DestinationEmailAddress` — L'indirizzo e-mail a cui ricevere un avviso quando è pianificata l'eliminazione di una chiave AWS KMS. • `SNSTopicName` — Il nome dell'argomento Amazon SNS. • `EnableRemediation` — Annullamento dell'eliminazione pianificata delle chiavi utilizzando un runbook Systems Manager. I valori consentiti sono `true` e `false`. • `AutomationAssumeRole` — L'Amazon Resource Name (ARN) del ruolo che consente all'automazione di Systems Manager di eseguire le azioni per tuo conto. Per ulteriori informazioni, consulta la sezione Autorizzazioni IAM richieste nella AWSConfigRemediation-CancelKeyDeletiondocumentazione. • `Capabilities` — Affinché AWS possa CloudFormation creare lo stack, è necessari	Sviluppatore, ingegnere della sicurezza

Operazione	Description	Competenze richieste
	o riconoscere esplicita mente che il modello di stack contiene determinate funzionalità.	

Confermare la sottoscrizione

Operazione	Description	Competenze richieste
Confermare la sottoscrizione.	Controlla la tua casella di posta elettronica e scegli Conferma abbonamento nel messaggio e-mail che ricevi da Amazon SNS. Si aprirà una finestra del browser Web che mostrerà una conferma dell'abbonamento e il tuo ID di abbonamento.	Sviluppatore, ingegnere della sicurezza

Risorse correlate

Riferimenti

- Creazione di una regola per un servizio AWS
- Creazione di un CloudWatch allarme Amazon per rilevare l'utilizzo di una chiave AWS KMS in attesa di eliminazione

Tutorial e video

- Come iniziare a usare Amazon EventBridge
- Approfondimento su Amazon EventBridge (AWS Online Tech Talks)

Workshop AWS

- Lavorare con EventBridge le regole

Informazioni aggiuntive

Il codice seguente fornisce esempi per estendere la soluzione per monitorare e notificare eventuali modifiche a qualsiasi servizio AWS. Gli esempi includono modelli predefiniti e modelli personalizzati. Per ulteriori informazioni, consulta Eventi e modelli di eventi in EventBridge.

```
EventPattern:
        source:
        - aws.kms
        detail-type:
        - AWS API Call via CloudTrail
        detail:
          eventSource:
          - kms.amazonaws.com
          eventName:
          - ScheduleKeyDeletion
```

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Identifica i bucket Amazon S3 pubblici utilizzando Security Hub (AWS Organizations CSPM)

Mourad Cherfaoui, Arun Chandapillai e Parag Nagwekar, Amazon Web Services

Riepilogo

Questo modello mostra come creare un meccanismo per identificare i bucket pubblici di Amazon Simple Storage Service (Amazon S3) nei tuoi account. AWS Organizations Il meccanismo funziona utilizzando i controlli dello standard AWS Foundational Security Best Practices (FSBP) per monitorare i bucket AWS Security Hub CSPM Amazon S3. Puoi utilizzare Amazon EventBridge per elaborare i risultati CSPM di Security Hub e poi pubblicarli su un argomento di Amazon Simple Notification Service (Amazon SNS). Le parti interessate della tua organizzazione possono iscriversi all'argomento e ricevere notifiche e-mail immediate sui risultati.

Per impostazione predefinita, i nuovi bucket Amazon S3 e i relativi oggetti non consentono l'accesso pubblico. Puoi utilizzare questo modello in scenari in cui devi modificare le configurazioni predefinite di Amazon S3 in base ai requisiti della tua organizzazione. Ad esempio, questo potrebbe essere uno scenario in cui hai un bucket Amazon S3 che ospita un sito Web pubblico o file che tutti gli utenti di Internet devono essere in grado di leggere dal tuo bucket Amazon S3.

Security Hub CSPM viene spesso implementato come servizio centrale per consolidare tutti i risultati di sicurezza, compresi quelli relativi agli standard di sicurezza e ai requisiti di conformità. Ce ne sono altri Servizi AWS che puoi usare per rilevare i bucket Amazon S3 pubblici, ma questo modello utilizza una distribuzione CSPM di Security Hub esistente con una configurazione minima.

Prerequisiti e limitazioni

Prerequisiti

- Una configurazione AWS multi-account con un account amministratore CSPM di Security Hub dedicato
- Security Hub (CSPM) e AWS Config, abilitato nel sistema Regione AWS che si desidera monitorare

> **Note**
>
> È necessario abilitare l'aggregazione tra regioni in Security Hub CSPM se si desidera monitorare più regioni da una singola regione di aggregazione.

- Autorizzazioni utente per accedere e aggiornare l'account amministratore CSPM di Security Hub, accesso in lettura a tutti i bucket Amazon S3 dell'organizzazione e autorizzazioni per disattivare l'accesso pubblico (se necessario)

Architecture

Il diagramma seguente mostra un'architettura per l'utilizzo di Security Hub CSPM per identificare i bucket Amazon S3 pubblici.

Il diagramma mostra il seguente flusso di lavoro:

1. Security Hub CSPM monitora la configurazione dei bucket Amazon S3 in tutti gli AWS Organizations account (incluso l'account amministratore) utilizzando i controlli S3.2 e S3.3 dello standard di sicurezza FSBP e rileva se un bucket è configurato come pubblico.
2. L'account amministratore CSPM di Security Hub accede ai risultati (inclusi quelli per S3.2 e S3.3) da tutti gli account dei membri.
3. Security Hub CSPM invia automaticamente tutti i nuovi risultati e tutti gli aggiornamenti ai risultati esistenti EventBridge come Security Hub CSPM Findings - Imported events. Ciò include gli eventi relativi ai risultati provenienti sia dall'account amministratore che da quello dei membri.
4. Una EventBridge regola filtra i risultati di S3.2 e S3.3 con un `ComplianceStatus` of`FAILED`, uno stato del workflow pari a e uno `RecordState` di`NEW`. `ACTIVE`
5. Le regole utilizzano i modelli di eventi per identificare gli eventi e inviarli a un argomento di Amazon SNS una volta che sono stati trovati.
6. Un argomento Amazon SNS invia gli eventi ai suoi abbonati (ad esempio tramite e-mail).
7. Gli analisti della sicurezza incaricati di ricevere le notifiche e-mail esaminano il bucket Amazon S3 in questione.
8. Se il bucket è approvato per l'accesso pubblico, l'analista della sicurezza imposta lo stato del flusso di lavoro del risultato corrispondente in Security Hub CSPM su. `SUPPRESSED` Altrimenti,

l'analista imposta lo stato su. `NOTIFIED` Ciò elimina le notifiche future per il bucket Amazon S3 e riduce il rumore delle notifiche.

9. Se lo stato del flusso di lavoro è impostato su`NOTIFIED`, l'analista della sicurezza esamina il risultato con il proprietario del bucket per determinare se l'accesso pubblico è giustificato e conforme ai requisiti di privacy e protezione dei dati. L'indagine porta alla rimozione dell'accesso pubblico al bucket o all'approvazione dell'accesso pubblico. In quest'ultimo caso, l'analista della sicurezza imposta lo stato del flusso di lavoro su. `SUPPRESSED`

> **Note**
>
> Il diagramma dell'architettura si applica sia alle implementazioni di aggregazione a regione singola che a livello interregionale. Negli account A, B e C del diagramma, Security Hub CSPM può appartenere alla stessa regione dell'account amministratore o appartenere a regioni diverse se l'aggregazione tra regioni è abilitata.

Tools (Strumenti)

- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. EventBridge fornisce un flusso di dati in tempo reale dalle tue applicazioni, dalle applicazioni SaaS (Software as a Service) e. Servizi AWS EventBridge indirizza i dati verso destinazioni come argomenti e AWS Lambda funzioni di Amazon SNS se i dati soddisfano le regole definite dall'utente.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail. I sottoscrittori ricevono tutti gli stessi messaggi pubblicati sugli argomenti ai quali sono hanno effettuato la sottoscrizione.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Security Hub CSPMfornisce una visione completa del tuo stato di sicurezza in. AWS Security Hub CSPM ti aiuta anche a controllare il tuo AWS ambiente rispetto agli standard e alle migliori pratiche del settore della sicurezza. Security Hub CSPM raccoglie dati sulla sicurezza da tutti Account AWS i servizi e dai prodotti partner di terze parti supportati, quindi aiuta ad analizzare le tendenze della sicurezza e identificare i problemi di sicurezza con la massima priorità.

Epiche

Configurazione degli account CSPM di Security Hub

Operazione	Description	Competenze richieste
Abilita Security Hub CSPM negli AWS Organizations account.	Per abilitare Security Hub CSPM negli account dell'organizzazione in cui desideri monitorare i bucket Amazon S3, consulta le linee guida tratte da Designazione di un account amministratore CSPM di Security Hub e Gestione degli account membro che appartengono a un'organizzazione nella documentazione CSPM di Security Hub.	Amministratore AWS
(Facoltativo) Abilita l'aggregazione tra regioni.	Se desideri monitorare i bucket Amazon S3 in più regioni da una singola regione, configura l'aggregazione tra regioni.	Amministratore AWS
Abilita i controlli S3.2 e S3.3 per lo standard di sicurezza FSBP.	È necessario abilitare i controlli S3.2 e S3.3 per lo standard di sicurezza FSBP. 1. Per abilitare i controlli S3.2, segui le istruzioni di [S3.2] I bucket S3 dovrebbero vietare l'accesso pubblico in lettura nella documentazione CSPM di Security Hub. 2. Per abilitare i controlli S3.3, segui le istruzioni fornite da	Amministratore AWS

Operazione	Description	Competenze richieste
	[3] I bucket S3 dovrebbero vietare l'accesso pubblico in scrittura nella documentazione CSPM di Security Hub.	

Configura l'ambiente

Operazione	Description	Competenze richieste
Configura l'argomento e l'abbonamento e-mail di Amazon SNS.	1. Accedi Console di gestione AWS e apri la console Amazon SNS. 2. Nel riquadro di navigazione scegliere Argomenti, quindi Crea nuovo argomento. 3. Per Tipo, scegliere Standard. 4. Per Nome, inserisci un nome per l'argomento (ad esempio, public-s3-buckets). 5. Scegli Create topic (Crea argomento). 6. Nella scheda Abbonamenti per il tuo argomento, scegli Crea abbonamento. 7. Per Protocol (Protocollo), selezionare Email (E-mail). 8. Per Endpoint, inserisci l'indirizzo email che riceverà le notifiche. È possibile utilizzare l'indirizzo e-mail di	Amministratore AWS

Operazione	Description	Competenze richieste
	un AWS amministratore, di un professionista IT o di un professionista Infosec. 9. Scegli Create Subscription (Crea sottoscrizione). Per creare sottoscrizioni e-mail aggiuntive, ripeti i passaggi 6—8 secondo necessità.	

Operazione	Description	Competenze richieste
Configura la EventBridge regola.	1. Apri la EventBridge console. 2. Nella sezione Inizia, seleziona EventBridge Regola, quindi scegli Crea regola. 3. Nella pagina di dettaglio Definisci regola, in Nome, inserisci un nome per la regola (ad esempio, public-s3-buckets). Scegli Next (Successivo). 4. Nella sezione Schema di eventi, scegli Modifica modello. 5. Copia il codice seguente, incollalo nell'editor del codice del modello di evento, quindi scegli Avanti. `{ "source": ["aws.securityhub"], "detail-type": ["Security Hub Findings - Imported"], "detail": { "findings": { "Compliance": { "Status": ["FAILED"] }, "RecordState": ["ACTIVE"], "Workflow": {`	Amministratore AWS

Operazione	Description	Competenze richieste
	<pre>"Status": ["NEW"] }, "ProductFields": { "ControlId": ["S3.2", "S3.3"] } } }</pre> 6. Nella pagina Seleziona destinazioni, per Seleziona una destinazione, seleziona l'argomento SNS come destinazione, quindi seleziona l'argomento che hai creato in precedenza. 7. Scegli Avanti, scegli di nuovo Avanti, quindi scegli Crea regola.	

risoluzione dei problemi

Problema	Soluzione
Ho un bucket Amazon S3 con accesso pubblico abilitato, ma non ricevo notifiche e-mail.	Ciò potrebbe essere dovuto al fatto che il bucket è stato creato in un'altra regione e l'aggregazione tra aree geografiche non è abilitata nell'account amministratore CSPM di Security Hub. Per risolvere questo problema, abilita l'aggregazione tra regioni o implement a la soluzione di questo pattern nella regione in cui risiede attualmente il tuo bucket Amazon S3.

Risorse correlate

- Che cos'è? AWS Security Hub CSPM (documentazione CSPM Security Hub)
- AWS Standard Foundational Security Best Practices (FSBP) (documentazione CSPM di Security Hub)
- AWS Security Hub CSPM script di abilitazione per più account (Labs)AWS
- Best practice di sicurezza per Amazon S3 (documentazione Amazon S3)

Informazioni aggiuntive

Flusso di lavoro per il monitoraggio dei bucket Amazon S3 pubblici

Il seguente flusso di lavoro illustra come monitorare i bucket Amazon S3 pubblici nella propria organizzazione. Il flusso di lavoro presuppone che tu abbia completato i passaggi indicati nell'argomento Configurazione di Amazon SNS e nella storia dell'abbonamento e-mail di questo modello.

1. Ricevi una notifica e-mail quando un bucket Amazon S3 è configurato con accesso pubblico.
   - Se il bucket è approvato per l'accesso pubblico, imposta lo stato del flusso di lavoro del risultato corrispondente `SUPPRESSED` nell'account amministratore CSPM di Security Hub. Ciò impedisce a Security Hub CSPM di emettere ulteriori notifiche per questo bucket e può eliminare gli avvisi duplicati.
   - Se il bucket non è approvato per l'accesso pubblico, imposta lo stato del flusso di lavoro del risultato corrispondente nell'account amministratore CSPM di Security Hub su. `NOTIFIED` Ciò impedisce a Security Hub CSPM di emettere ulteriori notifiche per questo bucket da Security Hub CSPM e può eliminare il rumore.
2. Se il bucket potrebbe contenere dati sensibili, disattiva immediatamente l'accesso pubblico fino al completamento della revisione. Se disattivi l'accesso pubblico, Security Hub CSPM modifica lo stato del flusso di lavoro in. `RESOLVED` Quindi, notifiche e-mail per il bucket stop.
3. Trova l'utente che ha configurato il bucket come pubblico (ad esempio, utilizzando AWS CloudTrail) e avvia una revisione. La revisione comporta la rimozione dell'accesso pubblico al bucket o l'approvazione dell'accesso pubblico. Se l'accesso pubblico è approvato, imposta lo stato del flusso di lavoro del risultato corrispondente su. `SUPPRESSED`

Inserisci e analizza i log di sicurezza AWS in Microsoft Sentinel

Ivan Girardi e Sebastian Wenzel, Amazon Web Services

Riepilogo

Questo modello descrive come automatizzare l'inserimento di log di AWS sicurezza, come log, dati Amazon Logs AWS CloudTrail , dati Amazon VPC Flow Logs e risultati di Amazon CloudWatch , in Microsoft Sentinel. GuardDuty Se la tua organizzazione utilizza Microsoft Sentinel come sistema di gestione delle informazioni e degli eventi di sicurezza (SIEM), questo ti aiuta a monitorare e analizzare centralmente i log per rilevare gli eventi relativi alla sicurezza. Non appena i log sono disponibili, vengono consegnati automaticamente a un bucket Amazon Simple Storage Service (Amazon S3) in meno di 5 minuti. Questo può aiutarti a rilevare rapidamente gli eventi di sicurezza nel tuo ambiente. AWS

Microsoft Sentinel inserisce CloudTrail i log in un formato tabulare che include il timestamp originale di quando l'evento è stato registrato. La struttura dei log importati abilita funzionalità di interrogazione utilizzando Kusto Query Language in Microsoft Sentinel.

Il pattern implementa una soluzione di monitoraggio e avviso che rileva gli errori di inserimento in meno di 1 minuto. Include anche un sistema di notifica che il SIEM esterno può monitorare. Si utilizza AWS CloudFormation per distribuire le risorse richieste nell'account di registrazione.

Destinatari

Questo modello è consigliato agli utenti che hanno esperienza con AWS Control Tower AWS Organizations, CloudFormation, AWS Identity and Access Management (IAM) e AWS Key Management Service (AWS KMS).

Prerequisiti e limitazioni

Prerequisiti

Di seguito sono riportati i prerequisiti per l'implementazione di questa soluzione:

- Account AWS Attivi gestiti come organizzazione all'interno di una AWS Control Tower landing zone AWS Organizations e facenti parte di essa. L'organizzazione deve includere un account dedicato per la registrazione. Per istruzioni, consulta Creazione e configurazione di un'organizzazione nella AWS Organizations documentazione.

- Un CloudTrail percorso che registra gli eventi per l'intera organizzazione e archivia i log in un bucket Amazon S3 nell'account di registrazione. Per istruzioni, consulta Creazione di un percorso per un'organizzazione.
- Nell'account di registrazione, le autorizzazioni per assumere un ruolo IAM esistente con le seguenti autorizzazioni:
  - Distribuisci le risorse definite nel modello fornito. CloudFormation
  - Implementa il modello fornito. CloudFormation
  - Modifica la politica delle AWS KMS chiavi se i log sono crittografati con una chiave gestita dal cliente.
- AWS Command Line Interface (AWS CLI), installato e configurato.
- Un account Microsoft Azure con un abbonamento per usare Microsoft Sentinel.
- Abilita e configura Microsoft Sentinel. Per istruzioni, consulta Abilitare Microsoft Sentinel e funzionalità e contenuti iniziali nella documentazione di Microsoft Sentinel.
- Soddisfa i prerequisiti per la configurazione del connettore Microsoft Sentinel S3.

Limitazioni

- Questa soluzione inoltra i log di sicurezza da un bucket Amazon S3 nell'account di registrazione a Microsoft Sentinel. Le istruzioni su come inviare i log ad Amazon S3 non vengono fornite esplicitamente.
- Questo modello fornisce istruzioni per il dispiegamento in una AWS Control Tower landing zone. Tuttavia, l'uso di non AWS Control Tower è richiesto.
- Questa soluzione è compatibile con un ambiente in cui il bucket di registrazione di Amazon S3 è limitato da politiche di controllo del servizio (SCPs), come Disallow Changes to Bucket Policy for Created Amazon AWS Control Tower S3 Bucket in Log Archive.
- Questo schema fornisce istruzioni per l'inoltro dei CloudTrail log, ma è possibile adattare questa soluzione per inviare altri log supportati da Microsoft Sentinel, come i log di Logs CloudWatch , Amazon VPC Flow Logs e. GuardDuty
- Le istruzioni utilizzano il per distribuire il modello, ma è anche possibile AWS CLI utilizzare il. CloudFormation Console di gestione AWS Per istruzioni, consulta Uso della AWS CloudFormation console. Se utilizzi la console per distribuire lo stack, distribuisci lo stack nello stesso Regione AWS bucket di registrazione.
- Questa soluzione implementa una coda Amazon Simple Queue Service (Amazon SQS) per inviare notifiche Amazon S3. La coda contiene messaggi con i percorsi degli oggetti caricati nel bucket

Amazon S3, non dati effettivi. La coda utilizza la crittografia SSE-SQS per proteggere il contenuto dei messaggi. Se desideri crittografare la coda SQS con SSE-KMS, puoi utilizzare una chiave KMS gestita dal cliente. Per ulteriori informazioni, consulta Encryption at rest in Amazon SQS.

Architecture

Questa sezione fornisce una panoramica di alto livello dell'architettura stabilita dal codice di esempio. Il diagramma seguente mostra le risorse distribuite nell'account di registrazione per importare i log da un bucket Amazon S3 esistente in Microsoft Sentinel.

Il diagramma dell'architettura mostra le seguenti interazioni con le risorse:

1. Nell'account di registrazione, Microsoft Sentinel assume un ruolo IAM tramite OpenID Connect (OIDC) per accedere ai log in uno specifico bucket Amazon S3 e nella coda Amazon SQS.
2. Amazon Simple Notification Service (Amazon SNS) e Amazon S3 vengono utilizzati per la crittografia. AWS KMS
3. Amazon S3 invia messaggi di notifica alla coda di Amazon SQS ogni volta che riceve nuovi log.
4. Microsoft Sentinel verifica la presenza di nuovi messaggi in Amazon SQS. La coda Amazon SQS utilizza la crittografia SSE-SQS. Il periodo di conservazione dei messaggi è impostato su 14 giorni.
5. Microsoft Sentinel estrae i messaggi dalla coda di Amazon SQS. I messaggi contengono il percorso degli oggetti Amazon S3 caricati. Microsoft Sentinel inserisce tali oggetti dal bucket Amazon S3 nell'account Microsoft Azure.
6. Un CloudWatch allarme monitora la coda di Amazon SQS. Se i messaggi non vengono ricevuti ed eliminati dalla coda di Amazon SQS entro 5 minuti, avvia una notifica Amazon SNS che invia un'e-mail.

AWS Control Tower ti aiuta a configurare la struttura delle unità organizzative (OU) di base e centralizza CloudTrail i log nell'account di registrazione. Implementa inoltre l'obbligo SCPs di proteggere il bucket di registrazione.

Abbiamo fornito l'architettura di destinazione in una AWS Control Tower landing zone, ma ciò non è strettamente necessario. In questo diagramma, le risorse dell'account di gestione riflettono una AWS Control Tower distribuzione e un CloudTrail percorso che registra gli eventi per l'intera organizzazione.

Questo modello si concentra sulla distribuzione delle risorse nell'account di registrazione. Se i log archiviati in Amazon S3 nella tua landing AWS Control Tower zone sono crittografati con una chiave KMS gestita dal cliente, devi aggiornare la policy delle chiavi per consentire a Microsoft Sentinel di decrittografare i log. In una AWS Control Tower landing zone, gestisci la politica delle chiavi dall'account di gestione, dove è stata creata la chiave.

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationti aiuta a configurare AWS le risorse, a fornirle in modo rapido e coerente e a gestirle durante tutto il loro ciclo di vita in tutte Account AWS le regioni.
- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue AWS risorse e delle applicazioni su cui esegui AWS in tempo reale.
- AWS Control Towerti aiuta a configurare e gestire un ambiente con AWS più account, seguendo le migliori pratiche.
- AWS Key Management Service (AWS KMS) consente di creare e controllare chiavi crittografiche per proteggere i dati.
- AWS Organizationsè un servizio di gestione degli account che consente di consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente.
- Amazon Simple Queue Service (Amazon SQS) fornisce una coda ospitata sicura, durevole e disponibile che ti aiuta a integrare e disaccoppiare sistemi e componenti software distribuiti.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri strumenti

- Microsoft Sentinel è un sistema SIEM nativo per il cloud che fornisce orchestrazione, automazione e risposta della sicurezza (SOAR).

Archivio di codici

Il codice per questo pattern è disponibile nel repository di GitHub Ingestisci e analizza i log AWS di sicurezza nel repository Microsoft Sentinel.

Best practice

- Segui il principio del privilegio minimo (documentazione IAM).
- Segui le migliori pratiche per gli AWS Control Tower amministratori (documentazione).AWS Control Tower
- Segui le AWS CloudFormation migliori pratiche (CloudFormation documentazione).
- Utilizza strumenti di analisi del codice, come cfn_nag, per scansionare i modelli generati. CloudFormation Lo strumento cfn_nag identifica potenziali problemi di sicurezza nei modelli cercando modelli. CloudFormation

Epiche

Connect Microsoft Sentinel ad Amazon S3

Operazione	Description	Competenze richieste
Preparare il connettore Microsoft Sentinel S3.	1. In Microsoft Sentinel, scegli Connettori dati. 2. Dalla galleria dei connettor i dati, scegli Amazon Web Services S3. **Note** Se non vedi il connettore, installa la soluzione Amazon Web Services dal Content Hub di Microsoft Sentinel. 3. Nel riquadro dei dettagli del connettore, scegli Apri pagina del connettore.	DevOps ingegnere, General AWS

Operazione	Description	Competenze richieste
	4. Nella sezione Configura zione, copia l'ID esterno. Questo ID ti servirà in un secondo momento.	

Implementa lo stack CloudFormation

Operazione	Description	Competenze richieste
Clonare il repository.	In una shell bash, inserisci il seguente comando. Questo clona i registri di AWS sicurezza di acquisizione e analisi nel repository Microsoft Sentinel. `git clone https://github.com/aws-samples/ingest-and-analyze-aws-security-logs-in-microsoft-sentinel.git`	DevOps ingegnere, General AWS
Assumi il ruolo IAM nell'account di registrazione.	Nell'account di registrazione, assumi il ruolo IAM che dispone delle autorizzazioni per distribuire lo stack. CloudFormation Per ulteriori informazioni sull'assunzione di un ruolo IAM in AWS CLI, consulta Utilizzare un ruolo IAM in. AWS CLI	DevOps ingegnere, General AWS

Operazione	Description	Competenze richieste
Implementa lo stack.	Per distribuire lo CloudFormation stack, inserisci il seguente comando, dove: • `<Bucket name>` è il nome del bucket Amazon S3 di registrazione. • `<Sentinel external ID>` è l'ID esterno del connettore Amazon S3 in Microsoft Sentinel. • `<Email address>` è un indirizzo e-mail valido a cui desideri ricevere notifiche. • `<Customer managed key ARN>` è l'Amazon Resource Name (ARN) della chiave KMS gestita dal cliente. Fornisci questo parametro solo se i log sono crittografati con una chiave KMS gestita dal cliente. • `<Suffix>` è un parametro opzionale per evitare conflitti tra i nomi delle risorse. • `<ARN for the OIDC provider>` è l'ARN del provider OIDC, se già esistente. Se non si fornisce questo parametro, CloudFormation crea il provider OIDC.	DevOps ingegnere, General AWS

Operazione	Description	Competenze richieste
	Important Se l' AWS organizzazione è monitorata con Microsoft Code Defender, il provider OIDC per Microsoft è già stato distribuito. È necessario fornire questo parametro e l'ARN del provider esistente.	
	`aws cloudformation deploy --stack-name cloudtrail-sentinel-integration \ --no-fail-on-empty-changeset \ --template-file template.yml \ --capabilities CAPABILITY_IAM CAPABILITY_NAMED_IAM CAPABILITY_AUTO_EXPAND \ --parameter-overrides \ ControlTowerS3BucketName="<Bucket name>" \ AzureWorkspaceID="<Sentinel external ID>" \`	

Operazione	Description	Competenze richieste
	`EmailAddress="<Email address>" \ KMSKeyArn="<Customer managed key ARN>" \ Suffix="<Suffix to avoid name conflicts>" \ OIDCProviderArn="<ARN for the OIDC provider>"`	
Copia gli output.	Dall'output dello CloudFormation stack, copia i valori per `SentinelRoleArn` e. `SentinelSQS` Questi valori verranno utilizzati in seguito per completare la configurazione in Microsoft Sentinel.	DevOps ingegnere, General AWS

Operazione	Description	Competenze richieste
Modifica la politica chiave.	Se non utilizzi una chiave KMS gestita dal cliente per crittografare i log nel bucket Amazon S3, puoi saltare questo passaggio. Se i log sono crittografati con una chiave KMS gestita dal cliente, modifica la politica delle chiavi per concedere a Microsoft Sentinel l'autoriz zazione a decrittografare i log. Di seguito è riportato un esempio di policy della chiave. Questo criterio di esempio consente l'accesso a più account se la chiave KMS si trova in un altro. Account AWS `{ "Version": "2012-10-17", "Id": "key-policy", "Statement": [ ... { "Sid": "Grant access to decrypt", "Effect": "Allow", "Principal": { "AWS": "<SentinelRoleArn>" }, "Action": "kms:Decrypt",`	DevOps ingegnere, General AWS

Operazione	Description	Competenze richieste
	<pre>"Resource": "<KeyArn>" }] }</pre>	

Configurare il connettore in Microsoft Sentinel

Operazione	Description	Competenze richieste
Completa la configurazione in Microsoft Sentinel.	1. In Microsoft Sentinel, scegli Connettori dati. 2. Dalla galleria dei connettori dati, scegli Amazon Web Services S3. 3. Nel riquadro dei dettagli del connettore, scegli Apri pagina del connettore. 4. Nella sezione Configurazione, procedi come segue: a. In Ruolo da aggiungere, inserisci il `SentinelRoleArn` valore che hai copiato. b. In SQS URL, inserisci il `SentinelSQS` valore che hai copiato. c. Nell'elenco della tabella di destinazione, scegli. `AWSCloudTrail` 5. Scegli Aggiungi connessione.	DevOps ingegnere

Operazione	Description	Competenze richieste
Invia notifiche di eventi Amazon S3 ad Amazon SQS.	Segui le istruzioni in Abilitazione e configurazione delle notifiche di eventi utilizzando la console Amazon S3 per configurare il bucket di registrazione Amazon S3 per inviare notifiche di eventi alla coda Amazon SQS. Se CloudTrail è stato configurato per l'intera organizzazione, i log in questo bucket hanno il prefisso, dove si trova l'ID dell'organizzazione. `<OrgID>/AWSLogs/<OrgID>/ <OrgID>` Per ulteriori informazioni, consulta Visualizzazione dei dettagli sull'organizzazione.	DevOps ingegnere, General AWS

Operazione	Description	Competenze richieste
Conferma che i log siano stati ingeriti.	1. Attendi che i log vengano inseriti in Microsoft Sentinel. Questo processo può richiedere diversi minuti. 2. In Microsoft Sentinel, apri la pagina Amazon S3 Data Connector, quindi procedi come segue: • Verifica che lo stato di Amazon S3 Data Connector sia. `Connected` • Controlla il volume dei dati nel grafico Dati ricevuti. Per ulteriori informazioni sull'ispezione dell'attività del connettore dati, vedere Connettore dati nella documentazione Microsoft.	DevOps ingegnere

Convalida la soluzione

Operazione	Description	Competenze richieste
Confronta CloudWatch e registra Sentinel.	Nella configurazione predefinita di AWS Control Tower, CloudTrail i log vengono inviati ad Amazon CloudWatch e archiviati nell'account di AWS Control Tower gestione. Per ulteriori informazioni, consulta Accesso e monitoraggio.	DevOps ingegnere, General AWS

Operazione	Description	Competenze richieste
	AWS Control Tower Utilizza la procedura seguente per confermare che i log vengano inseriti automaticamente in Microsoft Sentinel: 1. Apri la CloudWatch console. 2. Nel pannello di navigazione scegli Logs (Registri), quindi Logs Insights. 3. Per Seleziona gruppi di log, seleziona il gruppo di log in cui sono archiviati i log, ad CloudTrail esempio. `aws-controltower/CloudTrailLogs` 4. Nella casella dell'editor delle interrogazioni, immettete `fields eventID`. 5. Scegli Esegui query. 6. Scegli Esporta risultati, quindi scegli Copia tabella negli appunti (CSV). 7. Incolla i risultati in un editor di testo. 8. Modificare il formato dell'output in modo che possa essere utilizzato in una query di Microsoft Sentinel. Di seguito è riportato un esempio che	

Operazione	Description	Competenze richieste

	utilizza il linguaggio di interrogazione Kusto:
	
	`AWSCloudTrail`
	`| where AwsEventId in`
	`(`
	`'aa08b5fe-3bfb-391a-a14e-5fcebe14dab2',`
	`'9decd805-269c-451c-b75b-762f5dce59f9'`
	`)`
	
	9. In Microsoft Sentinel, apri la pagina Amazon S3 Data Connector. Accanto al grafico dei dati ricevuti, scegli Vai all'analisi dei log.
	10. Nella casella dell'editor di query, inserisci la query, quindi scegli Esegui.
	11. In Microsoft Sentinel e CloudWatch, verifica che il numero di voci sia lo stesso. Adatta l'intervallo di tempo, se necessario.	

Risorse correlate

AWS documentazione e risorse

- AWS CLI Command Reference (AWS CLI documentazione)
- Configurazione opzionale AWS KMS keys (AWS Control Tower documentazione)
- Crittografia inattiva in Amazon SQS (documentazione Amazon SQS)
- Come posso impedire ai destinatari della mailing list di annullare l'iscrizione a tutti i membri della lista dalle mie e-mail tematiche su Amazon SNS? (AWS Centro di conoscenza)

Documentazione Microsoft

- Connect Microsoft Sentinel ad Amazon Web Services per importare i dati del registro del servizio AWS
- Linguaggio di interrogazione Kusto in Microsoft Sentinel

Gestisci AWS Organizations le policy come codice utilizzando AWS CodePipeline Amazon Bedrock

André Cavalcante e Mariana Pessoa de Queiroz, Amazon Web Services

Riepilogo

Puoi utilizzare le politiche di autorizzazione per configurare e gestire centralmente AWS Organizations l'accesso ai principali e alle risorse nei tuoi account membro. Le politiche di controllo dei servizi (SCPs) definiscono le autorizzazioni massime disponibili per i ruoli e gli utenti AWS Identity and Access Management (IAM) dell'organizzazione. Le politiche di controllo delle risorse (RCPs) definiscono le autorizzazioni massime disponibili per le risorse dell'organizzazione.

Questo modello consente di gestire SCPs e utilizzare l'infrastruttura RCPs come codice (IaC) da implementare attraverso una pipeline di integrazione e distribuzione continue (CI/CD). Utilizzando AWS CloudFormation o Hashicorp Terraform per gestire queste politiche, è possibile ridurre l'onere associato alla creazione e al mantenimento di più politiche di autorizzazione.

Questo modello include le seguenti funzionalità:

- È possibile creare, eliminare e aggiornare le politiche di autorizzazione utilizzando i file manifest (`scp-management.jsonandrcp-management.json`).
- Si lavora con i guardrail anziché con le politiche. Definisci i tuoi guardrail e i loro obiettivi nei file manifest.
- La pipeline, che utilizza AWS CodeBuild e AWS CodePipeline, unisce e ottimizza i guardrail nei file manifest. Per ogni istruzione nel file manifest, la pipeline combina i guardrail in un unico SCP o RCP e quindi lo applica ai target definiti.
- AWS Organizations applica le politiche ai tuoi obiettivi. Un obiettivo può essere un' Account AWS unità organizzativa (OU), un ambiente (che è un gruppo di account o OUs definito nel `environments.json` file) o un gruppo di account che condividono un AWS tag.
- Amazon Bedrock legge i log della pipeline e riepiloga tutte le modifiche alle policy.
- La pipeline richiede un'approvazione manuale. L'approvatore può consultare il riepilogo esecutivo preparato da Amazon Bedrock, che lo aiuta a comprendere le modifiche.

Prerequisiti e limitazioni

Prerequisiti

- Molteplici Account AWS che vengono gestiti come organizzazione in. AWS Organizations Per ulteriori informazioni, vedere Creazione di un'organizzazione.
- Le funzionalità SCP e RCP sono abilitate in. AWS Organizations Per ulteriori informazioni, vedere Abilitazione di un tipo di policy.
- È installata la versione 1.9.8 o successiva di Terraform.
- Se non stai distribuendo questa soluzione tramite una pipeline Terraform, il file di stato Terraform deve essere archiviato in un bucket Amazon Simple Storage Service (Amazon S3) nella pipeline di gestione delle policy in Account AWS cui stai distribuendo.
- È installata la versione 3.13.3 o successiva di Python.

Limitazioni

- Non è possibile utilizzare questo pattern per gestire SCPs o RCPs che sono stati creati al di fuori di questa pipeline. CI/CD Tuttavia, è possibile ricreare le politiche esistenti tramite la pipeline. Per ulteriori informazioni, consulta Migrazione delle politiche esistenti nella pipeline nella sezione Informazioni aggiuntive di questo modello.
- Il numero di account e OUs le politiche di ciascun account sono soggetti alle quote e ai limiti di servizio per. AWS Organizations
- Questo modello non può essere utilizzato per configurare le politiche di gestione in AWS Organizations, ad esempio, le politiche di backup, le politiche dei tag, le politiche delle applicazioni di chat o le politiche dichiarative.

Architecture

Il diagramma seguente mostra il flusso di lavoro della pipeline di gestione delle policy e le risorse associate.

Il diagramma mostra il flusso di lavoro seguente:

1. Un utente esegue il commit delle modifiche ai file `scp-management.json` o `rcp-management.json` manifest nel ramo principale del repository remoto.
2. La modifica al `main` ramo avvia la pipeline. AWS CodePipeline

3. CodePipeline avvia il progetto. `Validate-Plan` CodeBuild Questo progetto utilizza uno script Python nel repository remoto per convalidare le politiche e i file manifest delle politiche. Questo CodeBuild progetto fa quanto segue:
   a. Verifica che i file manifest SCP e RCP contengano un'istruzione univoca IDs ()`Sid`.
   b. Utilizza gli script `scp-policy-processor/main.py` e `rcp-policy-processor/main.py` Python per concatenare i guardrail nella cartella guardrails in un'unica politica RCP o SCP. Combina guardrail che hanno lo stesso valore, e. `Resource` `Action` `Condition`
   c. Viene utilizzato AWS Identity and Access Management Access Analyzer per convalidare la politica finale ottimizzata. Se ci sono dei risultati, la pipeline si arresta.
   d. Crea `scps.json` e `rcps.json` archivia, che Terraform utilizza per creare risorse.
   e. Esegue il `terraform plan` comando, che crea un piano di esecuzione Terraform.
4. (Facoltativo) Il `Validate-Plan` CodeBuild progetto utilizza lo `bedrock-prompt/prompt.py` script per inviare un prompt ad Amazon Bedrock. Il prompt viene definito nel file. `bedrock-prompt/prompt.txt` Amazon Bedrock utilizza Anthropic Claude Sonnet 3.5 per generare un riepilogo delle modifiche proposte analizzando i log Terraform e Python.
5. CodePipeline utilizza un argomento Amazon Simple Notification Service (Amazon SNS) per notificare agli approvatori che le modifiche devono essere riviste. Se Amazon Bedrock ha generato un riepilogo delle modifiche, la notifica include questo riepilogo.
6. Un approvatore della politica approva l'azione in. CodePipeline Se Amazon Bedrock ha generato un riepilogo delle modifiche, l'approvatore può esaminarlo CodePipeline prima dell'approvazione.
7. CodePipeline avvia il progetto. `Apply` CodeBuild Questo progetto utilizza Terraform per applicare le modifiche RCP e SCP in. AWS Organizations

Il modello IaC associato a questa architettura implementa anche le seguenti risorse che supportano la pipeline di gestione delle politiche:

- Un bucket Amazon S3 per l'archiviazione di CodePipeline artefatti e script, come e `scp-policy-processor/main.py` `bedrock-prompt/prompt.py`
- Una chiave AWS Key Management Service (AWS KMS) che crittografa le risorse create da questa soluzione

Tools (Strumenti)

Servizi AWS

- Amazon Bedrock è un servizio di intelligenza artificiale completamente gestito che rende disponibili molti modelli di base ad alte prestazioni da utilizzare tramite un'API unificata.
- AWS CodeBuildè un servizio di compilazione completamente gestito che ti aiuta a compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per la distribuzione.
- AWS CodePipelineti aiuta a modellare e configurare rapidamente le diverse fasi di una versione del software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- AWS Organizationsè un servizio di gestione degli account che consente di consolidare più account Account AWS in un'organizzazione creata e gestita centralmente.
- AWS SDK per Python (Boto3)è un kit di sviluppo software che ti aiuta a integrare la tua applicazione, libreria o script Python con. Servizi AWS
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Altri strumenti

- HashiCorp Terraform è uno strumento IaC che ti aiuta a utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud.

Deposito di codici

Il codice per questo pattern è disponibile nel organizations-policy-pipeline GitHub repository. Di seguito sono riportati i file chiave contenuti nella `sample-repository` cartella:

- Nella `environments` cartella, `environments.json` contiene un elenco di ambienti. Gli ambienti sono un gruppo di obiettivi e possono contenere Account AWS IDs o unità organizzative (OUs).
- Nella `rcp-management` cartella:
  - La `guardrails` cartella contiene i parapetti individuali per il tuo RCPs
  - La `policies` cartella contiene l'individuo RCPs
  - Il file `rcp-management.json` manifest consente di gestire i guardrail RCP RCPs, full e gli obiettivi associati.
- Nella cartella: `scp-management`
  - La `guardrails` cartella contiene i parapetti individuali per il tuo SCPs
  - La `policies` cartella contiene l'individuo SCPs

  - Il file `scp-management.json` manifest consente di gestire i guardrail SCP SCPs, completi e gli obiettivi associati.
- La `utils` cartella contiene script che possono aiutarvi a migrare quelli attuali SCPs e RCPs a gestirli attraverso la pipeline. Per ulteriori informazioni, vedere la sezione Informazioni aggiuntive di questo modello.

Best practice

- Prima di configurare la pipeline, ti consigliamo di verificare di non aver raggiunto i limiti delle tue AWS Organizations quote.
- Ti consigliamo di utilizzare l'account di AWS Organizations gestione solo per le attività che devono essere eseguite in quell'account. Per ulteriori informazioni, consulta Best practice per l'account di gestione.

Epiche

Configura l'account di destinazione

Operazione	Description	Competenze richieste
Creare un repository .	Crea un repository da cui il tuo team addetto alle operazioni di sicurezza gestirà le politiche . Utilizza uno dei fornitori di repository di terze parti che AWS CodeConnections supportano.	DevOps ingegnere
Delegare l'amministrazione delle politiche.	Delega l'amministrazione delle AWS Organizations politiche all'account membro in cui stai distribuendo la pipeline. Per istruzioni, consulta Creare una politica di delega basata sulle risorse con. AWS Organizations Per una politica	Amministratore AWS

Operazione	Description	Competenze richieste
	di esempio, vedi Esempio di politica di delega basata sulle risorse nella sezione Informazioni aggiuntive di questo modello.	
(Facoltativo) Abilita il modello di base.	Se desideri generare riepiloghi delle modifiche alle policy, abilita l'accesso al modello di base Anthropic Claude 3.5 Sonnet in Amazon Bedrock nel luogo in Account AWS cui stai distribuendo la pipeline. Per istruzioni, consulta Aggiungere o rimuovere l'accesso ai modelli Amazon Bedrock Foundation.	Informazioni generali su AWS

Implementa le risorse per la pipeline

Operazione	Description	Competenze richieste
Clonare il repository.	Inserisci il seguente comando da cui clonare il organizations-policy-pipeline repository: GitHub `git clone https://github.com/aws-samples/organizations-policy-pipeline.git`	DevOps ingegnere
Definisci il tuo metodo di implementazione.	1. Nel repository clonato, apri il file. `variables.tf`	DevOps ingegnere

Operazione	Description	Competenze richieste
	2. Per `project_name`, inserisci il prefisso che desideri applicare ai nomi delle risorse distribuite. 3. Per `provider_type`, inserisci il provider dell'archivio remoto. Nel file vengono forniti valori validi. 4. Per `full_repository_name`, inserisci il nome del repository remoto. 5. Per `branch_name`, inserisci il nome del ramo Git che utilizzerai per distribuire le policy. Un push o un'unione in questo ramo avvia la pipeline. In genere, questo è il ramo principale. 6. Per `terraform_version`, inserisci la versione di Terraform che stai utilizzando. 7. Per `enable_bedrock`, inserisci `true` se desideri che Amazon Bedrock riassuma le modifiche. Inserisci `false` se non desideri generare un riepilogo delle modifiche. 8. Per `tags`, inserisci le coppie chiave-valore che desideri	

Operazione	Description	Competenze richieste
	assegnare come tag alle risorse distribuite. 9. Salvare e chiudere il file `variables.tf` .	

Implementa la pipeline.	1. Immettete il seguente comando per creare un piano e rivedere le modifiche:
	`terraform plan`
	2. Immettete il seguente comando per applicare il piano e creare l'infrast ruttura della pipeline:
	`terraform apply`
		DevOps ingegnere, Terraform

Connect il repository remoto.	Nel passaggio precedent e, Terraform ha creato una CodeConnections connessio ne al repository di terze parti. Nella console AWS Developer Tools, modifica lo stato della connessione da a`PENDING`. `AVAILABLE` Per istruzion i, consulta Aggiornare una connessione in sospeso.	AWS DevOps

Operazione	Description	Competenze richieste
Iscriviti all'argomento Amazon SNS.	Terraform ha creato un argomento Amazon SNS. Sottoscrivi un endpoint all'argomento e conferma l'iscrizione in modo che gli approvatori ricevano notifiche sulle azioni di approvazione in sospeso nella pipeline. Per istruzioni, consulta l'argomento Creazione di un abbonamento a un Amazon SNS.	Informazioni generali su AWS

Definisci i tuoi limiti e le tue politiche

Operazione	Description	Competenze richieste
Popola il repository remoto.	Dal repository clonato, copia il contenuto della `sample-repository` cartella nel tuo repository remoto. Sono incluse le cartelle`environments` ,`rcp-management` , `scp-management` e. `utils`	DevOps ingegnere
Definisci i tuoi ambienti.	1. Nella `environments` cartella, apri il `environments.json` file. Questo è il file in cui definisci l'obiettivo Account AWS e OUs per la tua RCPs mano SCPs. 2. Eliminate gli ambienti di esempio.	DevOps ingegnere

Operazione	Description	Competenze richieste
	3. Aggiungi i tuoi ambienti di destinazione nel seguente formato: `[ { "ID": "<environment-name>", "Target": [ "<ou-name>:<ou-id>", "<account-name>:<account-id>" ] } ]` Dove: • `<environment-name>` è il nome che assegni al gruppo OUs e agli account AWS. Puoi usare questo nome nel file manifest per definire dove vuoi applicare le tue policy. • `<ou-name>` è il nome dell'unità organizzativa di destinazione. • `<ou-id>`è l'ID dell'unità organizzativa di destinazione.	

Operazione	Description	Competenze richieste
	• `<account-name>` è il nome della destinazione Account AWS. • `<account-id>` è l'ID della destinazione Account AWS. Per esempi, consulta il repository del codice sorgente. 4. Salvare e chiudere il file `environments.json` .	

Operazione	Description	Competenze richieste
Definisci i tuoi guardrail.	1. Vai alla `rcp-management/guardrails` cartella nel tuo repository remoto. Questa è la cartella in cui definisci i guardrail per il tuo file manifest RCP. Ogni guardrail deve essere contenuto in un singolo file. I file Guardrail possono contenere una o più istruzioni. **Note** È possibile utilizzare lo stesso guardrail in più istruzioni nei file manifest per SCPs e. RCPs Se si modifica il guardrail, tutte le politiche che includono questo guardrail vengono influenzate. 2. Eliminate tutti i guardrail di esempio che sono stati copiati dal repository del codice sorgente. 3. Crea un nuovo file.json e assegnagli un nome descrittivo.	DevOps ingegnere

Operazione	Description	Competenze richieste
	4. Apri il file.json che hai creato. 5. Definite il guardrail nel seguente formato: `[ { "Sid": "<guardrail-name>", "Effect": "<effect-value>", "Action": [ "<action-name>" ], "Resource": "<resource-arn>", "Condition": { "<condition-operator>": { "<condition-key>": [ "<condition-value>" ] } } } ]` Dove: • `<guardrail-name>` è un nome univoco per il guardrail. Questo nome non può essere utilizzato per nessun altro guardrail .	

Operazione	Description	Competenze richieste
	• `<effect-value>` deve essere o. `Allow` `Deny` Per ulteriori informazioni, consulta Effect. • `<action-name>` deve essere un nome valido di un'azione supportata dal servizio. Per ulteriori informazioni, vedere Azione. • `<resource-arn>` è l'Amazon Resource Name (ARN) della risorsa a cui si applica il guardrail . Puoi anche usare caratteri jolly, come o. `*` `?` Per ulteriori informazioni, consulta Resource. • `<condition-operator>` è un operatore di condizione valido. Per ulteriori informazioni, vedere Operatori di condizione. • `<condition-key>` è una chiave di contesto globale valida o una chiave di contesto specifica del servizio. Per ulteriori informazioni, vedere Condizione. • `<condition-value>` è il valore specifico utilizzat	

Operazione	Description	Competenze richieste
	o in una condizione per valutare se è applicabile un guardrail. Per ulteriori informazioni, vedere Condizione. Ad esempio, guardrails RCP, consultate il repositor y del codice sorgente. 6. Salva e chiudi il file.json. 7. Ripeti questi passaggi per creare tutti i guardrail RCP necessari. 8. Ripeti questi passaggi nella `scp-management/guardrails` cartella per creare tutti i guardrail necessari per il tuo. SCPs Ad esempio, guardrail SCP, consultate il repository del codice sorgente.	

Operazione	Description	Competenze richieste
Definisci le tue politiche.	1. Vai alla `rcp-management/policies` cartella nel tuo repository remoto. Questa è la cartella in cui definisci le politiche complete per il tuo file manifest RCP. Ogni policy deve essere un singolo file. **Note** Se si modifica una politica in questa cartella, le modifiche alla politica influiscono su tutti gli account o a OUs cui viene applicata questa politica, come definito nel file manifesto. 2. Eliminare tutte le policy di esempio che sono state copiate dal repository del codice sorgente. 3. Crea un nuovo file.json e assegnagli un nome descrittivo. 4. Apri il file.json che hai creato. 5. Definisci l'RCP. Ad esempio RCPs, consulta il repositor	DevOps ingegnere

Operazione	Description	Competenze richieste
	y del codice sorgente o vedi Esempi di politiche di controllo delle risorse nella AWS Organizations documentazione. 6. Salva e chiudi il file.json. 7. Ripeti questi passaggi per crearne tutti RCPs quelli necessari. 8. Ripeti questi passaggi nella `scp-management/policies` cartella per crearne SCPs il numero necessario. Ad esempio SCPs, consulta il repository del codice sorgente o vedi gli esempi di policy di controllo dei servizi nella AWS Organizations documentazione.	

Usa il file manifest per gestire le politiche

Operazione	Description	Competenze richieste
Configura i file manifest.	1. Nella `rcp-management` cartella, apri il `rcp-management.json` file. Questo è il file in cui definisci quali guardrail RCP RCPs si applicano completamente agli ambienti di destinazione. Per un esempio di questo	DevOps ingegnere

Operazione	Description	Competenze richieste
	file, consultate il repository del codice sorgente. 2. Eliminate l'istruzione di esempio. 3. Aggiungere una nuova istruzione nel formato seguente: `[ { "SID": "<statement-name>", "Target": { "Type": "<target-type>", "ID": "<target-name>" }, "Guardrails": [ "<guardrail-name>" ], "Policy": "<policy-name>", "Comments": "<comment-text>" } ]` Dove: • `<statement-name>` è un nome univoco per l'istruzione. • `<target-type>` è il tipo di destinazione a cui si desidera applicare	

Operazione	Description	Competenze richieste

	la politica. I valori validi sono `Account`, 0U, `Environment` o `Tag`.
	• `<target-name>` è l'identificatore dell'obiettivo a cui si desidera applicare la politica. Immettere uno dei seguenti elementi:
	  • Per un Account AWS, inserisci l'identificatore come. `<account-name>:<account-id>`
	  • Per un'unità organizzativa, immettere l'identificatore come. `<0U-name>:<ou-id>`
	  • Per un ambiente, immettete il nome univoco definito nel `environments.json` file.
	  • Per un tag, inserisci la coppia chiave-valore come. `<tag-key>:<tag-value>`
	• `<guardrail-name>` è il nome univoco del guardrail RCP definito nella cartella. `rcp-management/guardrails` È possibile	

Operazione	Description	Competenze richieste
	aggiungere più guardrail in questo elemento. Puoi lasciare questo campo vuoto se non vuoi applicare un guardrail. • `<policy-name>` è il nome univoco dell'RCP definito nella `rcp-management/policies` cartella. È possibile aggiungere solo una politica in questo elemento. È possibile lasciare vuoto questo campo se non si desidera applicare una politica. • `<comment-text>` è una descrizione che è possibile inserire a scopo di documentazione. Questo campo non viene utilizzato durante l'elaborazione della pipeline. È possibile lasciare vuoto questo campo se non si desidera aggiungere un commento. 4. Ripeti questi passaggi per aggiungere tutte le istruzioni necessarie per la configurazione RCPs per la tua organizzazione.	

Operazione	Description	Competenze richieste
	5. Salvare e chiudere il file `rcp-management.json` . 6. Nella `scp-management` cartella, ripeti questi passaggi nel `scp-management.json` file. Questo è il file in cui definisci quali guardrail SCP SCPs si applicano completamente ai tuoi ambienti di destinazione. Per un esempio di questo file, consultate il repository del codice sorgente.	
Avvia la pipeline.	Conferma e invia le modifiche al ramo del repository remoto che hai definito nel `variables.tf` file. In genere, questo è il `main` ramo. La CI/CD pipeline si avvia automaticamente. Se sono presenti errori nella pipeline, consultate la sezione Risoluzione dei problemi di questo modello.	DevOps ingegnere

Operazione	Description	Competenze richieste
Approva le modifiche.	Una volta completato il `Validate-Plan` CodeBuild progetto, gli approvatori delle politiche ricevono una notifica tramite l'argomento Amazon SNS che hai configurato in precedenza. Esegui questa operazione: 1. Apri il messaggio di notifica. 2. Se disponibile, consulta il riepilogo delle modifiche alle politiche. 3. Segui le istruzioni riportate in Approvare o rifiutare un'azione di approvazione in. CodePipeline	AWS generale, responsab ile dell'approvazione delle politiche

Operazione	Description	Competenze richieste
Convalida la distribuzione.	1. Accedi alla AWS Organizations console con l'account di cui sei l'amministratore delegato. AWS Organizations 2. Nella pagina Criteri di controllo del servizio, verifica che siano elencati quelli SCPs che hai creato. 3. Scegliete un SCP gestito tramite la pipeline e confermate che si applichi agli obiettivi previsti. 4. Nella pagina delle politiche di controllo delle risorse, verifica che quelle RCPs che hai creato siano elencate. 5. Scegliete un RCP gestito tramite la pipeline e confermate che si applichi agli obiettivi previsti.	Informazioni generali su AWS

risoluzione dei problemi

Problema	Soluzione
Errori manifesti nei file nella `Validate-Plan` fase della pipeline	Se sono presenti errori nei file or, nell'output della pipeline viene visualizzato un messaggio «Errori della pipeline nella fase di convalida e pianificazione per i file manifest». `scp-management.json` `rcp-management.json` I possibili errori includono un nome di ambiente

Problema	Soluzione
	errato, campi o valori duplicati SIDs o non validi. Esegui questa operazione: 1. Segui le istruzioni in Visualizza i dettagli della build in. AWS CodeBuild 2. Nel registro di compilazione, trova l'errore di convalida. L'errore fornisce ulteriori informazioni su ciò che ha causato il fallimento della compilazione. 3. Aggiorna il file .json corrispondente. 4. Effettua il commit e invia il file aggiornato al repository remoto. La pipeline si riavvia. 5. Monitora lo stato per confermare che l'errore di convalida è stato risolto.

Problema	Soluzione
I risultati di IAM Access Analyzer nella `Validate-Plan` fase della pipeline	Un messaggio «Risultati di IAM Access Analyzer durante la fase di convalida e pianifica zione» viene visualizzato nell'output della pipeline in caso di errori nel guardrail o nelle definizioni delle policy. Questo modello utilizza IAM Access Analyzer per convalidare la policy finale. Esegui questa operazione: 1. Segui le istruzioni in Visualizza i dettagli della build in. AWS CodeBuild 2. Nel log di compilazione, trova l'errore di convalida di IAM Access Analyzer. L'errore fornisce ulteriori informazioni su cosa ha causato il fallimento della compilazione. Per ulteriori informazioni sui tipi di risultati , consulta IAM Policy Validation Check Reference. 3. Aggiorna il file.json corrispondente per il guardrail o la policy. 4. Effettua il commit e invia il file aggiornato al repository remoto. La pipeline si riavvia. 5. Monitora lo stato per confermare che l'errore di convalida è stato risolto.

Risorse correlate

- Riferimento agli elementi della policy JSON (documentazione IAM)
- Politiche di controllo delle risorse (AWS Organizations documentazione)
- Politiche di controllo dei servizi (AWS Organizations documentazione)
- Aggiungere o rimuovere l'accesso ai modelli Amazon Bedrock Foundation (documentazione Amazon Bedrock)
- Approva o rifiuta un'azione di approvazione in (documentazione) CodePipeline CodePipeline

Informazioni aggiuntive

Esempio di politica di delega basata sulle risorse

Di seguito è riportato un esempio di politica di delega basata sulle risorse per. AWS Organizations Consente all'account di amministrazione delegato di gestire SCPs e per l'organizzazione. RCPs Nel seguente esempio di policy, sostituiscilo `<MEMBER_ACCOUNT_ID>` con l'ID dell'account su cui stai distribuendo la pipeline di gestione delle policy.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DelegationToAudit",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::<MEMBER_ACCOUNT_ID>:root"
      },
      "Action": [
        "organizations:ListTargetsForPolicy",
        "organizations:CreatePolicy",
        "organizations:DeletePolicy",
        "organizations:AttachPolicy",
        "organizations:DetachPolicy",
        "organizations:DisablePolicyType",
        "organizations:EnablePolicyType",
        "organizations:UpdatePolicy",
        "organizations:DescribeEffectivePolicy",
        "organizations:DescribePolicy",
        "organizations:DescribeResourcePolicy"
      ],
      "Resource": "*"
    }
  ]
}
```

Migrazione delle politiche esistenti nella pipeline

Se ne hai esistenti SCPs o RCPs desideri migrare e gestire tramite questa pipeline, puoi usare gli script Python `sample-repository/utils` nella cartella del repository del codice. Questi script includono:

- `check-if-scp-exists-in-env.py`— Questo script verifica se una politica specificata si applica a qualsiasi destinazione in un ambiente specifico, definito nel `environments.json` file. Immettete il seguente comando per eseguire questo script:

```
python3 check-if-scp-exists-in-env.py \
   --policy-type <POLICY_TYPE> \
   --policy-name <POLICY_NAME> \
   --env-id <ENV_ID>
```

  Sostituisci quanto segue in questo comando:

  - <POLICY_TYPE> è `scp` o `rcp`
  - <POLICY_NAME>è il nome dell'SCP o dell'RCP
  - <ENV_ID>è l'ID dell'ambiente definito nel file `environments.json`
- `create-environments.py`— Questo script crea un file environments.json basato sull'ambiente corrente SCPs e RCPs su quello esistente. Sono escluse le politiche implementate tramite. AWS Control Tower Immettere il seguente comando per eseguire questo script, dove <POLICY_TYPE> è `scp` o: `rcp`

```
python create-environments.py --policy-type <POLICY_TYPE>
```

- `verify-policies-capacity.py`— Questo script controlla ogni ambiente definito per determinare la quantità di capacità residua per ogni quota AWS Organizations relativa alle politiche. L'utente definisce gli ambienti in cui archiviare il file in`environments.json`. Immettete il seguente comando per eseguire questo script, dove <POLICY_TYPE> è `scp` o`rcp`:

```
python verify-policies-capacity.py --policy-type <POLICY_TYPE>
```

Gestisci i set di AWS IAM Identity Center autorizzazioni come codice utilizzando AWS CodePipeline

André Cavalcante e Claison Amorim, Amazon Web Services

Riepilogo

AWS IAM Identity Center ti aiuta a gestire centralmente l'accesso Single Sign-On (SSO) a tutte le tue applicazioni. Account AWS Puoi creare e gestire le identità degli utenti in IAM Identity Center oppure puoi connettere una fonte di identità esistente, come un dominio Microsoft Active Directory o un provider di identità esterno (IdP). IAM Identity Center offre un'esperienza di amministrazione unificata per definire, personalizzare e assegnare un accesso granulare all'ambiente utilizzando set di autorizzazioni. AWS I set di autorizzazioni si applicano agli utenti e ai gruppi federati del tuo archivio di identità IAM Identity Center o del tuo IdP esterno.

Questo modello ti aiuta a gestire i set di autorizzazioni IAM Identity Center come codice nel tuo ambiente multi-account gestito come organizzazione in. AWS Organizations Con questo modello, puoi ottenere quanto segue:

- Creare, eliminare e aggiornare i set di autorizzazioni
- Crea, aggiorna o elimina le assegnazioni dei set di autorizzazioni a Target Account AWS, alle unità organizzative (OUs) o alla radice dell'organizzazione.

Per gestire le autorizzazioni e le assegnazioni di IAM Identity Center come codice, questa soluzione implementa una pipeline di integrazione e distribuzione continua (CI/CD) che utilizza e. AWS CodeBuild AWS CodePipeline Gestisci i set di autorizzazioni e le assegnazioni nei modelli JSON archiviati in un repository remoto. Quando EventBridge le regole di Amazon rilevano una modifica all'archivio o rilevano modifiche agli account nell'unità organizzativa di destinazione, avvia una AWS Lambda funzione. La funzione Lambda avvia la CI/CD pipeline che aggiorna i set di autorizzazioni e le assegnazioni in IAM Identity Center.

Prerequisiti e limitazioni

Prerequisiti

- Un ambiente multi-account gestito come organizzazione in. AWS Organizations Per ulteriori informazioni, vedere Creazione di un'organizzazione.

- IAM Identity Center, abilitato e configurato con una fonte di identità. Per ulteriori informazioni, consulta Getting Started nella documentazione di IAM Identity Center.
- Un account membro registrato come amministratore delegato per quanto segue: Servizi AWS
  - IAM Identity Center: per istruzioni, consulta Registrare un account membro nella documentazione di IAM Identity Center.
  - AWS Organizations — Per istruzioni, consulta Amministratore delegato per AWS Organizations. Questo account deve disporre delle autorizzazioni per elencare e descrivere gli account e. OUs

> **Note**
>
> È necessario utilizzare lo stesso account dell'amministratore delegato per entrambi i servizi.

- Autorizzazioni per distribuire gli AWS CloudFormation stack nell'account amministratore delegato di IAM Identity Center e nell'account di gestione dell'organizzazione. Per ulteriori informazioni, consulta Controllare l'accesso nella documentazione. CloudFormation
- Un bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) nell'account amministratore delegato di IAM Identity Center. Carichi il codice dell'artefatto in questo bucket. Per istruzioni, consulta Creazione di un bucket nella documentazione di Amazon S3.
- L'ID dell'account di gestione dell'organizzazione. Per istruzioni, vedi Ricerca del tuo Account AWS ID.
- Un repository nel tuo host di codice sorgente, ad esempio GitHub.

Limitazioni

- Questo modello non può essere utilizzato per gestire o assegnare set di autorizzazioni per ambienti con account singolo o per account che non sono gestiti come organizzazione in. AWS Organizations
- I nomi dei set di autorizzazioni IDs, l'assegnazione e i tipi principali di IAM Identity Center IDs non possono essere modificati dopo la distribuzione.
- Questo modello consente di creare e gestire autorizzazioni personalizzate. Non è possibile utilizzare questo modello per gestire o assegnare autorizzazioni predefinite.
- Questo modello non può essere utilizzato per gestire un set di autorizzazioni per l'account di gestione dell'organizzazione.

Architecture

Architettura Target

Il diagramma mostra il flusso di lavoro seguente:

1. Un utente apporta una delle seguenti modifiche:
   - Esegue il commit di una o più modifiche nell'archivio remoto, ad esempio GitHub
   - Modifica gli account nell'unità organizzativa in AWS Organizations
2. Se l'utente ha apportato una modifica all'archivio remoto al ramo principale, viene avviata la pipeline.

   Se l'utente ha modificato gli account nell'unità organizzativa, la `MoveAccount` EventBridge regola rileva la modifica e avvia una funzione Lambda nell'account di gestione dell'organizzazione.
3. La funzione Lambda avviata avvia la pipeline CI/CD in. CodePipeline
4. CodePipeline avvia il progetto. `TemplateValidation` CodeBuild Il `TemplateValidation` CodeBuild progetto utilizza uno script Python nel repository remoto per convalidare i modelli di set di autorizzazioni. CodeBuild convalida quanto segue:
   - I nomi dei set di autorizzazioni sono univoci.
   - L'istruzione di assegnazione IDs (`Sid`) è unica.
   - Definizioni delle politiche nel `CustomPolicy` parametro e valide. (Questa convalida utilizza AWS Identity and Access Management Access Analyzer.)
   - Gli Amazon Resource Names (ARNs) delle policy gestite sono validi.
5. Il gruppo di `PermissionSet` azioni del `Deploy` CodeBuild progetto utilizza AWS SDK per Python (Boto3) per eliminare, creare o aggiornare i set di autorizzazioni in IAM Identity Center. Sono interessati solo i set di autorizzazioni con il `SSOPipeline:true` tag. Tutti i set di autorizzazioni gestiti tramite questa pipeline hanno questo tag.
6. Il gruppo di `Assignments` azioni del `Deploy` CodeBuild progetto utilizza Terraform per eliminare, creare o aggiornare le assegnazioni in IAM Identity Center. I file di stato del backend Terraform sono archiviati in un bucket Amazon S3 nello stesso account.
7. CodeBuild aggiorna i set di autorizzazioni e le assegnazioni in IAM Identity Center.

Automazione e scalabilità

Poiché tutti i nuovi account in un ambiente con più account vengono spostati in un'unità organizzativa specifica nel AWS Organizations, questa soluzione viene eseguita automaticamente e concede i set di autorizzazioni richiesti a tutti gli account specificati nei modelli di assegnazione. Non sono necessarie automazioni o azioni di ridimensionamento aggiuntive.

In ambienti di grandi dimensioni, il numero di richieste API a IAM Identity Center potrebbe rallentare l'esecuzione di questa soluzione. Terraform e Boto3 gestiscono automaticamente il throttling per ridurre al minimo qualsiasi peggioramento delle prestazioni.

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita tra e. Account AWS Regioni AWS
- AWS CodeBuildè un servizio di compilazione completamente gestito che consente di compilare codice sorgente, eseguire test unitari e produrre artefatti pronti per l'implementazione.
- AWS CodePipelineti aiuta a modellare e configurare rapidamente le diverse fasi di una versione del software e ad automatizzare i passaggi necessari per rilasciare continuamente le modifiche al software.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altro modo. Account AWS
- AWS IAM Identity Centerti aiuta a gestire centralmente l'accesso Single Sign-On (SSO) a tutte le tue applicazioni e a quelle cloud. Account AWS
- AWS Organizationsè un servizio di gestione degli account che consente di consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente.
- AWS SDK per Python (Boto3)è un kit di sviluppo software che ti aiuta a integrare la tua applicazione, libreria o script Python con. Servizi AWS
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Deposito di codice

Il codice per questo pattern è disponibile nel repository aws-iam-identity-center-pipeline. La cartella templates nel repository include modelli di esempio sia per i set di autorizzazioni che per le

assegnazioni. Include anche AWS CloudFormation modelli per la distribuzione della CI/CD pipeline e AWS delle risorse negli account di destinazione.

Best practice

- Prima di iniziare a modificare il set di autorizzazioni e i modelli di assegnazione, si consiglia di pianificare i set di autorizzazioni per l'organizzazione. Considera quali dovrebbero essere le autorizzazioni, a quali account o OUs set di autorizzazioni dovrebbe applicarsi e quali principali di IAM Identity Center (utenti o gruppi) dovrebbero essere interessati dal set di autorizzazioni. I nomi dei set di autorizzazioni, le associazioni IDs e i tipi principali di IAM Identity Center IDs non possono essere modificati dopo la distribuzione.
- Rispetta il principio del privilegio minimo e concedi le autorizzazioni minime necessarie per eseguire un'attività. Per ulteriori informazioni, consulta le best practice relative alla concessione dei privilegi minimi e alla sicurezza nella documentazione AWS Identity and Access Management (IAM).

Epiche

Pianifica set di autorizzazioni e assegnazioni

Operazione	Description	Competenze richieste
Clonare il repository.	In una shell bash, inserisci il seguente comando. Questo clona il repository aws-iam-identity-center-pipeline da. GitHub `git clone https://github.com/aws-samples/aws-iam-identity-center-pipeline.git`	DevOps ingegnere
Definire i set di autorizzazioni.	1. Nel repository clonato, accedete alla `templates/permissionsets`	DevOps ingegnere

Operazione	Description	Competenze richieste
	cartella e aprite uno dei modelli disponibili. 2. Nel `Name` parametro, inserisci un nome per il set di autorizzazioni. Questo valore deve essere univoco e non può essere modificato dopo la distribuzione. 3. Nel `Description` parametro, descrivete brevemente il set di autorizzazioni, ad esempio il relativo caso d'uso. 4. Nel `SessionDuration` parametro, specifica per quanto tempo un utente può accedere a un Account AWS. Utilizza il formato di durata ISO-8601 (Wikipedia), ad `PT4H` esempio 4 ore. Se non viene definito alcun valore, l'impostazione predefinita in IAM Identity Center è 1 ora. 5. Nel `RelayState` parametro, specifica l'URL che fornisce un accesso rapido alla console più appropriata per il ruolo dell'utente. 6. Personalizza le politiche nel set di autorizzazioni. Tutti i seguenti parametri sono opzionali e possono essere	

Operazione	Description	Competenze richieste

	modificati dopo la distribuzione. È necessario utilizzare almeno uno dei parametri per definire le politiche nel set di autorizzazioni:
	• Nel `ManagedPolicies` parametro, inserisci le ARNs eventuali politiche AWS gestite che desideri assegnare.
	• Nel `CustomerManagedPolicies` parametro, inserisci i nomi di tutte le politiche gestite dai clienti che desideri assegnare. Non utilizzare l'ARN.
	• Nel `PermissionBoundary` parametro, effettuate le seguenti operazioni per assegnare un limite di autorizzazioni:
	  • Se utilizzi una policy AWS gestita come limite di autorizzazioni, in `PolicyType`, enter e in `AWSPolicy`, inserisci l'ARN della policy.
	  • Se utilizzi una policy gestita dai clienti come limite di autorizzazioni, in `PolicyType` `Customer`, enter e	

Operazione	Description	Competenze richieste
	in `Policy` inserisci il nome della policy. Non utilizzare l'ARN. • Nel `CustomPolicy` parametro, definite tutte le politiche personalizzate in formato JSON che desiderate assegnare . Per ulteriori informazioni sulla struttura delle politiche JSON, vedere Panoramica delle politiche JSON. 7. Salva e chiudi il modello di set di autorizzazioni. Si consiglia di salvare il file con un nome che corrisponda al nome del set di autorizzazioni. 8. Ripeti questo processo per creare tutti i set di autorizzazioni necessari per l'organizzazione ed eliminare tutti i modelli di esempio che non sono necessari.	

Operazione	Description	Competenze richieste
Definisci gli incarichi.	1. Nel repository clonato, accedete alla `templates/assignments` cartella, quindi apritela. `iam-identitycenter-assigments.json` Questo file descrive come si desidera assegnare i set di autorizzazioni a o. Account AWS OUs 2. Nel `SID` parametro, inserite un identificatore per l'assegnazione. Questo valore deve essere univoco e non può essere modificato dopo la distribuzione. 3. Nel `Target` parametro, definisci gli account o le organizzazioni a cui desideri applicare il set di autorizzazioni. I valori validi sono account IDs OUs, o`root`. `root`assegna il set di autorizzazioni a tutti gli account dei membri dell'organizzazione, escluso l'account di gestione. Inserisci i valori tra virgolette doppie e separa più valori con virgole. Account IDs e OUs dovrebbe seguire lo schema: `{{account_name}}:{{account_id}}` o. `{{ou_name}}:`	DevOps ingegnere

Operazione	Description	Competenze richieste
	`{{ou_id}}` Se desideri assegnare le autorizzazioni in modo ricorsivo a nested OUs, usa il pattern OU con un carattere jolly alla fine. Ad esempio: `{{ou_name}}:{{ou_id}}:*` 4. Nel `PrincipalType` parametro, inserisci il tipo di principale IAM Identity Center che sarà interessato dal set di autorizzazioni. I valori validi sono `USER` e `GROUP`. Questo valore non può essere modificato dopo la distribuzione. 5. Nel `PrincipalID` parametro, inserisci il nome dell'utente o del gruppo nell'archivio di identità di IAM Identity Center che sarà interessato dal set di autorizzazioni. Questo valore non può essere modificato dopo la distribuzione. 6. Nel `PermissionSetName` parametro, inserisci il nome del set di autorizzazioni che desideri assegnare. 7. Ripeti i passaggi da 2 a 6 per creare tutte le assegnazioni necessarie	

Operazione	Description	Competenze richieste
	in questo file. In genere, esiste un'assegnazione per ogni set di autorizza zioni. Eliminare eventuali assegnazioni di esempio che non sono obbligatorie. 8. Salvare e chiudere il file `iam-identitycenter -assigments.json` .	

Distribuisci i set di autorizzazioni e le assegnazioni

Operazione	Description	Competenze richieste
Implementa le risorse nell'acco unt amministratore delegato di IAM Identity Center.	1. Nell'account amministr atore delegato IAM Identity Center, apri la console.C loudFormation 2. Implementa il modello. `iam-identitycenter -pipeline.yaml` Assegna allo stack un nome chiaro e descritti vo e aggiorna i parametri come indicato. Per istruzion i, consultate Creazione di uno stack nella documenta zione. CloudFormation	DevOps ingegnere
Distribuisci le risorse nell'acco unt AWS Organizations di gestione.	1. Accedi all'account di gestione dell'organizzazion e. 2. Apri la AWS CloudForm ation console.	DevOps ingegnere

Operazione	Description	Competenze richieste
	3. Nella barra di navigazione, scegli il nome del file attualmente visualizzato Regione AWS. Quindi scegli la `us-east-1` regione. Questa regione è necessaria per consentire alla `MoveAccount` EventBridge regola di rilevare AWS CloudTrail gli eventi associati ai cambiamenti dell'organizzazione. 4. Implementa il `iam-identitycenter-organization` modello. Assegna allo stack un nome chiaro e descrittivo e aggiorna i parametri come indicato. Per istruzioni, consulta Creazione di uno stack nella documentazione. CloudFormation	
Completa la configurazione del repository remoto.	Modifica lo stato della AWS CodeConnections connessione da `PENDING` a `AVAILABLE`. Questa connessione è stata creata quando hai distribuito lo CloudFormation stack. Per istruzioni, consulta Aggiornare una connessione in sospeso nella documentazione. CodeConnections	DevOps ingegnere

Operazione	Description	Competenze richieste
Carica i file nell'archivio remoto.	Carica tutti i file che hai scaricato dal `aws-samples` repository e modificato nei passaggi precedenti nell'archivio remoto. Le modifiche al `main` ramo avviano la pipeline, che crea o aggiorna i set di autorizzazioni e le assegnazioni.	DevOps ingegnere

Aggiornamento dei set di autorizzazioni e delle assegnazioni

Operazione	Description	Competenze richieste
Aggiorna i set di autorizzazioni e le assegnazioni.	Quando la EventBridge regola `MoveAccount` Amazon rileva modifiche agli account dell'organizzazione, la CI/CD pipeline si avvia automaticamente e aggiorna i set di autorizzazioni. Ad esempio, se aggiungi un account a un'unità organizzativa specificata nel file JSON delle assegnazioni, la CI/CD pipeline applicherà il set di autorizzazioni al nuovo account. Se desideri modificare i set di autorizzazioni e le assegnazioni distribuiti, aggiorna i file JSON e quindi esegui il commit nell'archivio remoto.	DevOps ingegnere

Operazione	Description	Competenze richieste
	Tieni presente quanto segue quando utilizzi la CI/CD pipeline per gestire i set di autorizzazioni e le associazioni precedentemente distribuiti: • Se si modifica il nome di un set di autorizzazioni, la CI/CD pipeline elimina il set di autorizzazioni originale e ne crea uno nuovo. • Questa pipeline gestisce solo i set di autorizzazioni che hanno il tag. `SSOPipeline:true` • È possibile avere più set di autorizzazioni e modelli di assegnazione nella stessa cartella del repository. • Se si elimina un modello, la pipeline elimina l'assegnazione o il set di autorizzazioni. • Se elimini un intero blocco JSON di assegnazione, la pipeline elimina l'assegnazione da IAM Identity Center. • Non è possibile eliminare un set di autorizzazioni assegnato a un. Account AWS Innanzitutto, è necessario annullare	

Operazione	Description	Competenze richieste
	l'assegnazione del set di autorizzazioni.	

risoluzione dei problemi

Problema	Soluzione
Errori di accesso negato	Conferma di disporre delle autorizzazioni necessarie per distribuire i CloudFormation modelli e le risorse definite al loro interno. Per ulteriori informazioni, consulta Controllo dell'accesso nella CloudFormation documenta zione.
Errori della pipeline nella fase di convalida	Questo errore viene visualizzato se sono presenti errori nel set di autorizzazioni o nei modelli di assegnazione. 1. In CodeBuild, visualizza i dettagli della build. 2. Nel registro di compilazione, trova l'errore di convalida che fornisce ulteriori informazioni sulla causa del fallimento della compilazione. 3. Aggiorna il set di autorizzazioni o i modelli di assegnazione, quindi esegui il commit nel repository. 4. La CI/CD pipeline riavvia il progetto. CodeBuild Monitora lo stato per confermare che l'errore di convalida è stato risolto.

Risorse correlate

- Set di autorizzazioni (documentazione IAM Identity Center)

Gestisci le credenziali con AWS Secrets Manager

Durga Prasad Cheepuri, Amazon Web Services

Riepilogo

Questo modello illustra come utilizzare AWS Secrets Manager per recuperare dinamicamente le credenziali del database per un'applicazione Java Spring.

In passato, quando creavi un'applicazione personalizzata che recuperava informazioni da un database, solitamente dovevi integrare le credenziali (il segreto) per accedere al database direttamente nell'applicazione. Quando era il momento di ruotare le credenziali, era necessario dedicare tempo all'aggiornamento dell'applicazione per utilizzare le nuove credenziali e quindi distribuire l'applicazione aggiornata. Se aveste più applicazioni che condividono le credenziali e non si aggiorna una di esse, l'applicazione fallirebbe. A causa di questo rischio, molti utenti hanno scelto di non ruotare regolarmente le proprie credenziali, il che di fatto sostituiva un rischio con un altro.

Secrets Manager consente di sostituire le credenziali codificate nel codice (comprese le password) con una chiamata API per recuperare il segreto a livello di codice. Questo aiuta a garantire che il segreto non possa essere compromesso da qualcuno che sta esaminando il codice, perché il segreto semplicemente non c'è. Puoi anche configurare Secrets Manager per ruotare automaticamente il segreto in base a una pianificazione specificata. Ciò consente di sostituire i segreti a lungo termine con segreti a breve termine, il che aiuta a ridurre significativamente il rischio di compromissione. Per ulteriori informazioni, consulta la documentazione di AWS Secrets Manager.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS con accesso a Secrets Manager
- Un'applicazione Java Spring

Architecture

Stack di tecnologia di origine

- Un'applicazione Java Spring con codice che accede a un database, con credenziali DB gestite dal file application.properties.

Stack tecnologico Target

- Un'applicazione Java Spring con codice che accede a un database, con credenziali DB gestite in Secrets Manager. Il file application.properties contiene i segreti di Secrets Manager.

Integrazione di Secrets Manager con un'applicazione

Tools (Strumenti)

- Secrets Manager: AWS Secrets Manager è un servizio AWS che semplifica la gestione dei segreti. I segreti possono essere le credenziali dei database, le password, le chiavi API di terza parte e anche le parti di testo arbitrario. È possibile archiviare e controllare l'accesso a questi segreti centralmente utilizzando la console Secrets Manager, l'interfaccia a riga di comando (CLI) di Secrets Manager o l'API Secrets Manager e. SDKs

Epiche

Conserva segreti in Secrets Manager

Operazione	Description	Competenze richieste
Memorizza le credenziali del DB come segreto in Secrets Manager.	Archivia Amazon Relational Database Service (Amazon RDS) o altre credenziali DB come segreto in Secrets Manager seguendo i passaggi descritti in Creazione di un segreto nella documentazione di Secrets Manager.	Amministratore di sistema
Imposta le autorizzazioni per l'applicazione Spring per accedere a Secrets Manager.	Imposta le autorizzazioni appropriate in base a come l'applicazione Java Spring utilizza Secrets Manager. Per controllare l'accesso al segreto, crea una policy	Amministratore di sistema

Operazione	Description	Competenze richieste
	basata sulle informazioni fornite nella documentazione di Secrets Manager, nelle sezioni Utilizzo di politiche basate sull'identità (IAM Policies) e ABAC for Secrets Manager e Utilizzo di politiche basate sulle risorse per Secrets Manager. Segui i passaggi indicati nella sezione Recupero del valore segreto nella documentazione di Secrets Manager.	

Aggiorna l'applicazione Spring

Operazione	Description	Competenze richieste
Aggiungi dipendenze JAR per usare Secrets Manager.	Vedi la sezione Informazioni aggiuntive per i dettagli.	Sviluppatore Java
Aggiungi i dettagli del segreto all'applicazione Spring.	Aggiorna il file applicati on.properties con il nome segreto, gli endpoint e la regione AWS. Per un esempio, consulta la sezione Informazioni aggiuntive.	Sviluppatore Java
Aggiorna il codice di recupero delle credenziali DB in Java.	Nell'applicazione, aggiorna il codice Java che recupera le credenziali del DB per recuperare quei dettagli da Secrets Manager. Per esempio di codice, consultat	Sviluppatore Java

Operazione	Description	Competenze richieste
	e la sezione Informazioni aggiuntive.	

Risorse correlate

- Documentazione di AWS Secrets Manager
- Utilizzo di politiche basate sull'identità (IAM Policies) e ABAC for Secrets Manager
- Utilizzo di politiche basate sulle risorse per Secrets Manager
- Codice di esempio

Informazioni aggiuntive

Aggiungere dipendenze JAR per l'utilizzo di Secrets Manager

Maven:

```
<groupId>com.amazonaws</groupId>
    <artifactId>aws-java-sdk-secretsmanager</artifactId>
    <version>1.11. 355 </version>
```

Gradle:

```
compile group: 'com.amazonaws', name: 'aws-java-sdk-secretsmanager', version:
 '1.11.355'
```

Aggiornamento del file application.properties con i dettagli del segreto

```
spring.aws.secretsmanager.secretName=postgres-local
spring.aws.secretsmanager.endpoint=secretsmanager.us-east-1.amazonaws.com
spring.aws.secretsmanager.region=us-east-1
```

Aggiornamento del codice di recupero delle credenziali DB in Java

```
String  secretName  =  env.getProperty("spring.aws.secretsmanager.secretName");
String  endpoints  =  env.getProperty("spring.aws.secretsmanager.endpoint");
```

```
String  AWS Region  =  env.getProperty("spring.aws.secretsmanager.region");
AwsClientBuilder.EndpointConfiguration  config  =  new
  AwsClientBuilder.EndpointConfiguration(endpoints, AWS Region);
AWSSecretsManagerClientBuilder  clientBuilder  =
  AWSSecretsManagerClientBuilder.standard();
clientBuilder.setEndpointConfiguration(config);
AWSSecretsManager  client  =  clientBuilder.build();

ObjectMapper  objectMapper  =  new  ObjectMapper();

JsonNode  secretsJson  =  null;

ByteBuffer  binarySecretData;

GetSecretValueRequest  getSecretValueRequest  =  new
  GetSecretValueRequest().withSecretId(secretName);

GetSecretValueResult  getSecretValueResponse  =  null;

try  {
     getSecretValueResponse  =  client.getSecretValue(getSecretValueRequest);
    }

catch  (ResourceNotFoundException  e)  {
     log.error("The requested secret "  +  secretName  +  " was not found");
    }

catch  (InvalidRequestException  e)  {
     log.error("The request was invalid due to: "  +  e.getMessage());
     }

catch  (InvalidParameterException  e)  {
     log.error("The request had invalid params: "  +  e.getMessage());
     }
if  (getSecretValueResponse  ==  null)  {
     return  null;
     }  // Decrypted secret using the associated KMS key // Depending on whether the
 secret was a string or binary, one of these fields will be populated


String secret = getSecretValueResponse.getSecretString();

if (secret != null) {
     try {
```

```
                secretsJson  =  objectMapper.readTree(secret);
           }

     catch  (IOException  e)  {
                log.error("Exception while retrieving secret values: "  +
  e.getMessage());
           }
}

else  {
     log.error("The Secret String returned is null");

     return null;

     }
     String  host  =  secretsJson.get("host").textValue();
     String  port  =  secretsJson.get("port").textValue();
     String  dbname  =  secretsJson.get("dbname").textValue();
     String  username  =  secretsJson.get("username").textValue();
     String  password  =  secretsJson.get("password").textValue();
}
```

Monitora ElastiCache i cluster Amazon per la crittografia a riposo

Abhishek Agawane, Amazon Web Services

Riepilogo

Amazon ElastiCache è un servizio Amazon Web Services (AWS) che fornisce una soluzione di caching ad alte prestazioni, scalabile ed economica per la distribuzione di un archivio dati in memoria o un ambiente di cache nel cloud. Recupera i dati da archivi di dati in memoria ad alta velocità e bassa latenza. Questa funzionalità lo rende una scelta popolare per casi d'uso in tempo reale come memorizzazione nella cache, archivi di sessioni, giochi, servizi geospaziali, analisi in tempo reale e accodamento. ElastiCache offre archivi dati Redis e Memcached, entrambi con tempi di risposta inferiori al millisecondo.

La crittografia dei dati aiuta a impedire agli utenti non autorizzati di leggere i dati sensibili disponibili sui cluster Redis e sui sistemi di storage cache associati. Ciò include i dati salvati su supporti persistenti, noti come dati a riposo, e i dati che possono essere intercettati mentre viaggiano attraverso la rete tra client e server di cache, noti come dati in transito.

È possibile abilitare la crittografia a riposo per ElastiCache (Redis OSS) quando si crea un gruppo di replica, impostando il parametro su. `AtRestEncryptionEnabled` **`true`** Quando questo parametro è abilitato, crittografa il disco durante le operazioni di sincronizzazione, backup e swap e crittografa i backup archiviati in Amazon Simple Storage Service (Amazon S3). Non è possibile abilitare la crittografia a riposo su un gruppo di replica esistente. Quando si crea un gruppo di replica, è possibile abilitare la crittografia a riposo in questi due modi:

- Scegliendo l'opzione Default, che utilizza la crittografia a riposo gestita dal servizio.
- Utilizzando una chiave gestita dal cliente e fornendo l'ID della chiave o l'Amazon Resource Name (ARN) da AWS Key Management Service (AWS KMS).

Questo modello fornisce un controllo di sicurezza che monitora le chiamate API e genera un evento Amazon EventBridge Events sull'`CreateReplicationGroup`operazione. Questo evento chiama una AWS Lambda funzione che esegue uno script Python. La funzione ottiene l'ID del gruppo di replica dall'input JSON dell'evento ed esegue i seguenti controlli per determinare se esiste un cluster non crittografato:

- Verifica se la chiave esiste. `AtRestEncryptionEnabled`
- Se `AtRestEncryptionEnabled` esiste, controlla il valore per vedere se lo è`true`.

- Se il `AtRestEncryptionEnabled` valore è impostato su`false`, imposta una variabile che tiene traccia delle violazioni e invia un messaggio di violazione a un indirizzo e-mail fornito, utilizzando una notifica Amazon Simple Notification Service (Amazon SNS).

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS.
- Un bucket S3 per caricare il codice Lambda fornito.
- Un indirizzo email a cui desideri ricevere le notifiche di violazione.
- ElastiCache registrazione abilitata, per l'accesso a tutti i log delle API.

Limitazioni

- Questo controllo investigativo è regionale e deve essere implementato in ogni area Regione AWS che si desidera monitorare.
- Il controllo supporta i gruppi di replica in esecuzione in un cloud privato virtuale (VPC).
- Il controllo supporta i gruppi di replica che eseguono i seguenti tipi di nodi:
  - R7g, R6gd, R6g, R5, R4, R3
  - M7g, 6g, M5, M4, M3
  - T4g, T3, T2
  - C7gn

Versioni del prodotto

- Supporta la versione 3.2.6 o successiva ElastiCache (Redis OSS) e Valkey 7.2 o successiva

Architecture

Architettura del workflow

1. L'utente avvia un gruppo di ElastiCache replica tramite Console di gestione AWS, the AWS Command Line Interface (AWS CLI) o una chiamata API.

2. ElastiCache genera EventBridge eventi quando viene chiamata l'`CreateReplicationGroup` API.
3. Una EventBridge regola attiva e chiama la funzione Lambda per il controllo della conformità.
4. La funzione Lambda elabora l'evento e verifica se la crittografia a riposo è abilitata nel cluster. ElastiCache
5. Se viene rilevata una violazione della crittografia, la funzione Lambda pubblica un messaggio di notifica su un argomento SNS.
6. Amazon SNS invia una notifica e-mail agli amministratori in merito alla violazione della conformità della crittografia.

Automazione e scalabilità

- Se lo utilizzi AWS Organizations, puoi AWS CloudFormation StackSetsutilizzarlo per distribuire questo modello in più account che desideri monitorare.

Tools (Strumenti)

Servizi AWS

- Amazon ElastiCache semplifica la configurazione, la gestione e la scalabilità di ambienti di cache in memoria distribuiti in. Cloud AWS Fornisce una cache in memoria ad alte prestazioni, ridimensionabile ed economica, eliminando al contempo la complessità associata all'implementazione e alla gestione di un ambiente di cache distribuito. ElastiCache funziona con entrambi i motori Redis e Memcached.
- AWS CloudFormationti aiuta a modellare e configurare AWS le tue risorse, a fornirle in modo rapido e coerente e a gestirle per tutto il loro ciclo di vita. È possibile utilizzare un modello per descrivere le risorse e le relative dipendenze e lanciarle e configurarle insieme come uno stack, anziché gestire le risorse singolarmente. Puoi gestire ed eseguire il provisioning degli stack su più sistemi. Account AWS Regioni AWS
- Amazon EventBridge offre un flusso quasi in tempo reale di eventi di sistema che descrivono i cambiamenti nelle AWS risorse. EventBridge viene a conoscenza dei cambiamenti operativi man mano che si verificano e intraprende le azioni correttive necessarie, inviando messaggi per rispondere all'ambiente, attivando funzioni, apportando modifiche e acquisendo informazioni sullo stato.

- AWS Lambdaè un servizio di elaborazione che supporta l'esecuzione di codice senza fornire o gestire server. Lambda esegue il codice solo quando necessario e passa automaticamente da poche richieste al giorno a migliaia al secondo. Verrà addebitato soltanto il tempo di calcolo consumato e non verrà addebitato alcun costo quando il codice non è in esecuzione.
- Amazon SNS coordina e gestisce l'invio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail. I sottoscrittori ricevono tutti gli stessi messaggi pubblicati sugli argomenti ai quali sono hanno effettuato la sottoscrizione.

Codice

Il codice per questo modello è disponibile nell'archivio di crittografia GitHub Monitor Amazon ElastiCache Clusters for at-rest. Consulta la sezione Epics per informazioni su come utilizzare i file nel repository.

Best practice

Distribuzione

- Assicurati che AWS CloudTrail stia registrando le chiamate ElastiCache API prima di distribuire questo controllo.
- Si tratta di un controllo regionale; distribuisci il controllo in ogni Regione AWS luogo in cui lo utilizzi. ElastiCache
- Convalida la soluzione negli dev/test ambienti prima di implementarla in produzione.

Sicurezza

- Per un maggiore controllo sulle chiavi di crittografia, utilizza chiavi KMS gestite dal cliente.
- Rivedi le autorizzazioni AWS Identity and Access Management (IAM) per garantire l'accesso con il minimo privilegio per il ruolo di esecuzione Lambda.
- Imposta gli avvisi per i messaggi nella coda delle lettere morte.

Operazioni

- Imposta la conservazione dei log appropriata per bilanciare le esigenze di conformità con i costi.
- Regola la concorrenza riservata di Lambda per adattarla in base ElastiCache alla frequenza di creazione.

- Sottoscrivi più indirizzi e-mail ad Amazon SNS per le notifiche del team.

Monitoraggio

- Controlla gli CloudWatch allarmi di Amazon per assicurarti che le soglie di allarme corrispondano alle tue esigenze operative.
- Monitora regolarmente la durata dell'esecuzione delle metriche Lambda e i tassi di errore.
- Verifica regolarmente le violazioni per esaminare le notifiche di conformità alla crittografia.

Epiche

Implementa il controllo di sicurezza

Operazione	Description	Competenze richieste
Scarica il codice da GitHub.	Clona o scarica il repository del codice da. GitHub Il repository contiene i file e. `index.py` `elasticache_encryption_at_rest.yml`	Architetto del cloud
Crea pacchetti di distribuzione Lambda.	Crea due file.zip dal codice Python: • Crea `ElastiCache-EncryptionAtRest.zip` , che contiene. `index.py` • Utilizza il seguente comando: `zip ElastiCache-EncryptionAtRest.zip index.py`	Architetto del cloud

Operazione	Description	Competenze richieste
Carica il codice in un bucket S3.	1. Crea un nuovo bucket S3 o usa un bucket S3 esistente per caricare il codice Lambda. 2. Comprimi il codice Lambda (`index.py`) e assegnagli un nome. `ElastiCache-EncryptionAtRest.zip` 3. Carica il file.zip nel bucket S3. Questo bucket deve trovarsi nella Regione AWS stessa cartella delle risorse che desideri valutare.	Architetto del cloud
Implementa il CloudFormation modello.	Apri la CloudFormation console nello Regione AWS stesso bucket S3 e distribuisci il `elasticache_encryption_at_rest.yml` file fornito nel repository del codice. Nella prossima epopea, fornisci i valori per i parametri del modello.	Architetto del cloud

Completa i parametri nel CloudFormation modello

Operazione	Description	Competenze richieste
Fornisci il nome del bucket S3.	Inserisci il nome del bucket S3 che hai creato o selezionato nella prima epic. Questo bucket S3 contiene il file.zip per il codice Lambda e deve	Architetto del cloud

Operazione	Description	Competenze richieste
	essere nello stesso Regione AWS del CloudFormation modello e della risorsa che verranno valutati.	
Fornisci la chiave S3.	Fornisci la posizione del file.zip del codice Lambda nel tuo bucket S3, senza barre iniziali (ad esempio o). `ElasticCache-EncryptionAtRest.zip` `controls/ElasticCache-EncryptionAtRest.zip`	Architetto del cloud
Fornisci un indirizzo email.	Fornisci un indirizzo email attivo a cui desideri ricevere le notifiche di violazione.	Architetto del cloud
Specificare un livello di registrazione.	Specificare il livello di registrazione e la verbosità. • `Info`indica messaggi informativi dettagliati sullo stato di avanzamento dell'applicazione e deve essere utilizzato solo per il debug. • `Error`indica eventi di errore che potrebbero comunque consentire all'applicazione di continuare a funzionare. • `Warning`indica situazioni potenzialmente dannose.	Architetto del cloud

Confermare la sottoscrizione

Operazione	Description	Competenze richieste
Conferma l'iscrizione via e-mail.	Quando il CloudFormation modello viene distribuito correttamente, invia un messaggio di iscrizione all'indirizzo e-mail fornito. Per ricevere notifiche, devi confermare questa sottoscrizione e-mail.	Architetto del cloud

risoluzione dei problemi

Problema	Soluzione
Funzione Lambda non attivata	Sintomo: nessun accesso CloudWatch dopo aver creato o modificato i cluster. ElastiCache Soluzioni: • Verifica lo stato della EventBridge regola: `aws events describe-rule --name <rule-name>` • Verifica che la policy basata sulle risorse di Lambda consenta l'invocazione. EventBridge • Verifica che il modello di evento corrisponda alle chiamate API. EventBridge ElastiCache
Nessuna notifica via e-mail	Sintomo: la funzione Lambda viene eseguita correttamente, ma non si ricevono notifiche e-mail. Soluzioni:

Problema	Soluzione
	• Conferma l'abbonamento ad Amazon SNS tramite il link di conferma via e-mail. • Controlla le cartelle spam o posta indesider ata per le e-mail di Amazon SNS. • Verifica l'Amazon Resource Name (ARN) per l'argomento Amazon SNS nella variabile di ambiente Lambda: `OUTBOUND_TOPIC_ARN` • Esegui il test manuale di Amazon SNS: `aws sns publish --topic-arn <arn> --message "Test"`
Problemi a livello di autorizzazioni	Sintomo: accesso negato agli errori nei registri delle funzioni CloudWatch Lambda. Soluzioni: • Verifica che il ruolo di esecuzione Lambda disponga delle autorizzazioni di ElastiCache lettura richieste. • Verifica che la policy AWS KMS chiave includa il ruolo di esecuzione Lambda. • Assicurati che la policy tematica di Amazon SNS consenta la pubblicazione a Lambda.

Risorse correlate

- Crea uno stack dalla CloudFormation console (documentazione)CloudFormation
- Crittografia a riposo in ElastiCache (Redis OSS) (documentazione) ElastiCache

Monitoraggio dell'attività dell'utente root IAM

JJ Sung e Mostefa Brougui, Amazon Web Services

Riepilogo

Ogni account Amazon Web Services (AWS) ha un utente root. Come best practice di sicurezza per AWS Identity and Access Management (IAM), consigliamo di utilizzare l'utente root per completare le attività che solo l'utente root può eseguire. Per l'elenco completo, consulta Attività che richiedono credenziali utente root nella AWS Account Management Reference Guide. Poiché l'utente root ha pieno accesso a tutte le tue risorse AWS e ai dati di fatturazione, ti consigliamo di non utilizzare questo account e di monitorarlo per eventuali attività, che potrebbero indicare che le credenziali dell'utente root sono state compromesse.

Utilizzando questo modello, configuri un'architettura basata sugli eventi che monitora l'utente root IAM. Questo modello imposta una hub-and-spoke soluzione che monitora più account AWS, gli account spoke, e centralizza la gestione e il reporting in un unico account, l'account hub.

Quando vengono utilizzate le credenziali utente root IAM, Amazon CloudWatch e AWS CloudTrail registrano l'attività rispettivamente nel log e nel trail. Nell'account spoke, una EventBridge regola Amazon invia l'evento al bus degli eventi centrale nell'account hub. Nell'account hub, una EventBridge regola invia l'evento a una funzione AWS Lambda. La funzione utilizza un argomento Amazon Simple Notification Service (Amazon SNS) che notifica l'attività dell'utente root.

In questo modello, utilizzi un CloudFormation modello AWS per distribuire i servizi di monitoraggio e gestione degli eventi negli account spoke. Si utilizza un modello HashiCorp Terraform per distribuire i servizi di gestione degli eventi e di notifica nell'account dell'hub.

Prerequisiti e limitazioni

Prerequisiti

1. Autorizzazioni per distribuire risorse AWS nel tuo ambiente AWS.
2. Autorizzazioni per distribuire set di stack CloudFormation . Per ulteriori informazioni, consulta Prerequisiti per le operazioni relative agli stack set (documentazione). CloudFormation
3. Terraform installato e pronto all'uso. Per ulteriori informazioni, consulta Get Started — AWS (documentazione Terraform).
4. Una traccia esistente in ogni account spoke. Per ulteriori informazioni, consulta Getting started with AWS CloudTrail (CloudTrail documentazione).

5. Il percorso è configurato per inviare eventi a CloudWatch Logs. Per ulteriori informazioni, vedere Invio di eventi ai CloudWatch registri (CloudTrail documentazione).
6. I tuoi account hub and spoke devono essere gestiti da AWS Organizations.

Architecture

Il diagramma seguente illustra gli elementi costitutivi dell'implementazione.

1. Quando vengono utilizzate le credenziali dell'utente root IAM, CloudWatch CloudTrail registrate l'attività rispettivamente nel log e nel trail.
2. Nell'account spoke, una EventBridge regola invia l'evento al bus degli eventi centrale nell'account hub.
3. Nell'account hub, una EventBridge regola invia l'evento a una funzione Lambda.
4. La funzione Lambda utilizza un argomento di Amazon SNS che notifica l'attività dell'utente root.

Tools (Strumenti)

Servizi AWS

- AWS ti CloudFormation aiuta a configurare le risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita su account e regioni AWS.
- AWS ti CloudTrail aiuta a controllare la governance, la conformità e il rischio operativo del tuo account AWS.
- Amazon CloudWatch Logs ti aiuta a centralizzare i log di tutti i tuoi sistemi, applicazioni e servizi AWS in modo da poterli monitorare e archiviare in modo sicuro.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, funzioni AWS Lambda, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altri account AWS.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di elaborazione che ti aiuta a eseguire codice senza dover fornire o gestire server. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di calcolo che utilizzi.

- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.

Altri strumenti e servizi

- Terraform è un'applicazione CLI per il provisioning e la gestione dell'infrastruttura e delle risorse cloud utilizzando il codice, sotto forma di file di configurazione.

Archivio di codice

Il codice sorgente e i modelli di questo pattern sono disponibili in un GitHub repository. Questo modello fornisce due modelli:

- Un modello Terraform contenente le risorse distribuite nell'account hub
- Un CloudFormation modello che distribuisci come istanza stack set negli account spoke

Il repository ha la seguente struttura generale.

```
.
 |__README.md
 |__spoke-stackset.yaml
 |__hub.tf
 |__root-activity-monitor-module
     |__main.tf  # contains Terraform code to deploy resources in the Hub account
     |__iam      # contains IAM policies JSON files
         |__ lambda-assume-policy.json          # contains trust policy of the IAM role
 used by the Lambda function
         |__ lambda-policy.json                 # contains the IAM policy attached to
 the IAM role used by the Lambda function
     |__outputs  # contains Lambda function zip code
```

La sezione Epics fornisce step-by-step istruzioni per la distribuzione dei modelli.

Epiche

Distribuisci le risorse sull'account hub

Operazione	Description	Competenze richieste
Clona il repository di codice di esempio.	1. Apri il repository AWS IAM Root User Activity Monitor. 2. Nella scheda Codice, sopra l'elenco dei file, scegli Codice, quindi copia l'URL HTTPS. 3. In un'interfaccia a riga di comando, modificate la directory di lavoro nella posizione in cui desiderate archiviare i file di esempio. 4. Immetti il comando seguente: `git clone <repoURL>`	Informazioni generali su AWS
Aggiorna il modello Terraform.	1. Recupera l'ID della tua organizzazione. Per istruzioni, consulta Visualizzazione dei dettagli di un'organizzazione dall'account di gestione (documentazione di AWS Organizations). 2. Nel repository clonato, apri. `hub.tf` 3. Aggiorna quanto segue con i valori appropriati per il tuo ambiente:	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	• `OrganizationId` — Aggiungi l'ID della tua organizzazione. • `SNSTopicName` — Aggiungi un nome per l'argomento Amazon SNS. • `SNSSubscriptions` — Aggiungi l'e-mail a cui inviare le notifiche di Amazon SNS. • `Region`— Aggiungi il codice della regione AWS in cui stai distribuendo le risorse. Ad esempio, `eu-west-1` . • `Tags`— Aggiungi i tuoi tag. Per ulteriori informazioni, consulta Tagging AWS resources (AWS General Reference). 4. Salvare e chiudere il file `hub.tf`.	

Operazione	Description	Competenze richieste

Distribuisci le risorse nell'account dell'hub AWS.	1. Nell'interfaccia a riga di comando Terraform, vai alla cartella principale del repository clonato, quindi inserisci il seguente comando.
	`terraform init && terraform plan`
	2. Rivedi l'output e conferma di voler creare le risorse descritte.
	3. Inserire il seguente comando.
	`terraform apply`
	4. Quando richiesto, conferma la distribuzione `yes` inserendo.
		Informazioni generali su AWS

Distribuisci le risorse sui tuoi account Spoke

Operazione	Description	Competenze richieste
Implementa il CloudFormation modello.	1. Accedere alla Console di gestione AWS e aprire la console CloudFormation . 2. Dal riquadro di navigazione, scegli StackSets. 3. Nella parte superiore della StackSetspagina, scegli Crea StackSet.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	4. In Autorizzazioni, scegli Autorizzazioni gestite dal servizio. CloudFormation configura automaticamente le autorizzazioni necessari e per la distribuzione sugli account di destinazione gestiti da AWS Organizat ions. 5. In Prerequisito - Prepara modello, scegli Il modello è pronto. 6. In Specificare modello, scegli Carica un file modello. 7. Scegli il file, quindi, nel repository clonato, seleziona. `spoke-sta ckset.yaml` 8. Scegli Next (Successivo). 9. Nella pagina Specificare StackSet i dettagli, inserisci un nome per il set di stack. 10. In Parametri, inserisci l'ID dell'account hub, quindi scegli Avanti. 11. Nella pagina Configura StackSet opzioni, sotto Tag, aggiungi i tuoi tag. 12. In Configurazione di esecuzione, scegli Inattivo, quindi scegli Avanti.	

Operazione	Description	Competenze richieste
	13. Nella pagina Imposta le opzioni di distribuzione, specifica le unità organizzative e le regioni in cui desideri distribuire lo stack set, quindi scegli Avanti. 14. Nella pagina di revisione, seleziona Riconosco che AWS CloudFormation potrebbe creare risorse IAM, quindi scegli Invia. CloudFormation inizia a distribuire il tuo set di stack. Per ulteriori informazioni e istruzioni, consulta Creare un set di stack (document azione)CloudFormation .	

(Facoltativo) Prova le notifiche

Operazione	Description	Competenze richieste
Usa le credenziali dell'utente root.	1. Accedi a un account spoke o all'account hub utilizzando le credenziali dell'utente root. 2. Verifica che l'account e-mail che hai specificato riceva la notifica di Amazon SNS.	Informazioni generali su AWS

Risorse correlate

- Best practice di sicurezza (documentazione IAM)
- Lavorare con StackSets (CloudFormation documentazione)
- Inizia (documentazione Terraform)

Informazioni aggiuntive

Amazon GuardDuty è un servizio di monitoraggio continuo della sicurezza che analizza ed elabora i log per identificare attività impreviste e potenzialmente non autorizzate nel tuo ambiente AWS. In alternativa a questa soluzione, se l'hai abilitata GuardDuty, può avvisarti quando sono state utilizzate le credenziali dell'utente root. Il GuardDuty risultato è`Policy:IAMUser/RootCredentialUsage`, e la gravità predefinita è Bassa. Per ulteriori informazioni, consulta la sezione Gestione dei GuardDuty risultati di Amazon.

Invia una notifica quando viene creato un utente IAM

Mansi Suratwala e Sergiy Shevchenko, Amazon Web Services

Riepilogo

Su Amazon Web Services (AWS), puoi utilizzare questo modello per distribuire un CloudFormation modello AWS per ricevere notifiche automaticamente quando vengono creati utenti AWS Identity and Access Management (IAM).

Utilizzando IAM, puoi gestire l'accesso ai servizi e alle risorse AWS in modo sicuro. Puoi creare e gestire utenti e gruppi AWS e utilizzare le autorizzazioni per consentire e negare a tali utenti e gruppi l'accesso alle risorse AWS.

Il CloudFormation modello crea un evento Amazon CloudWatch Events e una funzione AWS Lambda. L'evento utilizza AWS CloudTrail per monitorare qualsiasi utente IAM creato nell'account AWS. Se viene creato un utente, l'evento CloudWatch Events avvia una funzione Lambda, che ti invia una notifica Amazon Simple Notification Service (Amazon SNS) che ti informa dell'evento di creazione di un nuovo utente.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo
- Un CloudTrail percorso AWS creato e distribuito

Limitazioni

- Il CloudFormation modello AWS deve essere distribuito `CreateUser` solo per.

Architecture

Stack tecnologico Target

- IAM
- AWS CloudTrail
- CloudWatch Eventi Amazon

- AWS Lambda
- Amazon Simple Storage Service (Amazon S3)
- Amazon SNS

Architettura Target

Automazione e scalabilità

Puoi utilizzare il CloudFormation modello AWS più volte per diverse regioni e account AWS. Devi eseguirlo solo una volta in ogni regione o account. Per automatizzare la distribuzione su più account, usa AWS CloudFormation StackSets. Il CloudFormation modello sarà in grado di distribuire tutte le risorse richieste in ogni account.

Tools (Strumenti)

Strumenti

- IAM: AWS Identity and Access Management (IAM) è un servizio Web che ti aiuta a controllare in modo sicuro l'accesso alle risorse AWS. Utilizza IAM per controllare chi è autenticato (accesso effettuato) e autorizzato (dispone di autorizzazioni) per l'utilizzo di risorse.
- AWS CloudFormation: AWS ti CloudFormation aiuta a modellare e configurare le tue risorse Amazon Web Services in modo da dedicare meno tempo alla gestione di tali risorse e più tempo alle applicazioni eseguite in AWS. Crei un modello che descrive tutte le risorse AWS che desideri e si CloudFormation occupa del provisioning e della configurazione di tali risorse per te.
- AWS CloudTrail: AWS ti CloudTrail aiuta a gestire la governance, la conformità e il controllo operativo e dei rischi del tuo account AWS. Le azioni intraprese da un utente, un ruolo o un servizio AWS vengono registrate come eventi in CloudTrail. Gli eventi includono azioni intraprese nella Console di gestione AWS, nell'interfaccia a riga di comando AWS e in AWS SDKs e APIs.
- Amazon CloudWatch Events: Amazon CloudWatch Events offre un near-real-time flusso di eventi di sistema che descrivono i cambiamenti nelle risorse AWS.
- AWS Lambda: AWS Lambda è un servizio di elaborazione che supporta l'esecuzione di codice senza effettuare il provisioning o la gestione di server. Lambda esegue il codice solo quando è necessario e si dimensiona automaticamente, da poche richieste al giorno a migliaia al secondo.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è uno storage per Internet. È possibile utilizzare Amazon S3 per memorizzare e recuperare qualsiasi volume di dati, in qualunque momento e da qualunque luogo tramite il Web.

- Amazon SNS — Amazon Simple Notification Service (Amazon SNS) è un servizio gestito che fornisce il recapito di messaggi tramite Lambda, HTTP, e-mail, notifiche push mobili e messaggi di testo mobili (SMS).

Codice

Un file.zip del progetto è disponibile come allegato.

Epiche

Crea il bucket S3 per lo script Lambda

Operazione	Description	Competenze richieste
Definisci il bucket S3.	Apri la console Amazon S3 e scegli o crea un bucket S3. Questo bucket S3 ospiterà il file.zip con codice Lambda. Il nome del bucket S3 non può contenere barre iniziali.	Architetto del cloud

Carica il codice Lambda nel bucket S3

Operazione	Description	Competenze richieste
Carica il codice Lambda.	Carica il file.zip con codice Lambda fornito nella sezione Allegati nel bucket S3 che hai definito.	Architetto del cloud

Implementa il modello CloudFormation

Operazione	Description	Competenze richieste
Implementa il CloudFormation modello.	Sulla CloudFormation console, distribuisci il CloudFormation	Architetto del cloud

Operazione	Description	Competenze richieste
	`createIAMuser.yaml` modello fornito come allegato a questo modello. Nella prossima epopea, fornisci i valori per i parametri del modello.	

Completa i parametri nel CloudFormation modello

Operazione	Description	Competenze richieste
Fornisci il nome del bucket S3.	Inserisci il nome del bucket S3 che hai creato o scelto nella prima epic.	Architetto del cloud
Fornisci la chiave S3.	Fornisci la posizione del file.zip del codice Lambda nel tuo bucket S3, senza barre iniziali (ad esempio,) . `<directory>/<file-name>.zip`	Architetto del cloud
Fornisci un indirizzo email.	Fornisci un indirizzo e-mail attivo per ricevere le notifiche di Amazon SNS.	Architetto del cloud
Definisci il livello di registraz ione.	Definisci il livello e la frequenza di registrazione per la tua funzione Lambda. `Info`indica messaggi informati vi dettagliati sullo stato di avanzamento dell'appl icazione. `Error`indica eventi di errore che potrebbero comunque consentire all'appli	Architetto del cloud

Operazione	Description	Competenze richieste
	cazione di continuare a funzionare. `Warning`indica situazioni potenzialmente dannose.	

Confermare la sottoscrizione

Operazione	Description	Competenze richieste
Confermare la sottoscrizione.	Quando il modello viene distribuito correttamente, invia un messaggio e-mail di sottoscrizione all'indirizzo e-mail fornito. Per ricevere notifiche, è necessario confermare questa sottoscri zione e-mail.	Architetto del cloud

Risorse correlate

- Creating a trail
- Creazione di un bucket S3
- Caricamento di file su un bucket S3
- Distribuzione di un modello CloudFormation
- Creazione di un utente IAM
- Creazione di una regola CloudWatch Events che si attiva su una chiamata API AWS utilizzando AWS CloudTrail

Allegati

Per accedere a contenuti aggiuntivi associati a questo documento, decomprimi il seguente file: attachment.zip

Impedisci l'accesso a Internet a livello di account utilizzando una politica di controllo del servizio

Sergiy Shevchenko, Sean O'Sullivan e Victor Mazeo Whitaker, Amazon Web Services

Riepilogo

Organizations spesso desidera limitare l'accesso a Internet alle risorse degli account che dovrebbero rimanere private. In questi account, le risorse nei cloud privati virtuali (VPCs) non devono accedere a Internet in alcun modo. Molte organizzazioni scelgono un'architettura di ispezione centralizzata. Per il traffico est-ovest (da VPC a VPC) in un'architettura di ispezione centralizzata, è necessario assicurarsi che gli account spoke e le relative risorse non abbiano accesso a Internet. Per il traffico nord-sud (uscita Internet e locale), si desidera consentire l'accesso a Internet solo tramite il VPC di ispezione.

Questo modello utilizza una policy di controllo dei servizi (SCP) per impedire l'accesso a Internet. È possibile applicare questo SCP a livello di account o unità organizzativa (OU). L'SCP limita la connettività Internet impedendo quanto segue:

- Creazione o collegamento di un IPv4 gateway IPv6 Internet che consenta l'accesso diretto a Internet al VPC
- Creazione o accettazione di una connessione peering VPC che potrebbe consentire l'accesso indiretto a Internet tramite un altro VPC
- Creazione o aggiornamento di una AWS Global Acceleratorconfigurazione che potrebbe consentire l'accesso diretto a Internet alle risorse VPC

Prerequisiti e limitazioni

Prerequisiti

- Una o più Account AWS gestite come organizzazione in AWS Organizations.
- Tutte le funzionalità sono abilitate in AWS Organizations.
- SCPs sono abilitate nell'organizzazione.
- Autorizzazioni per:
  - Accedere all'account di gestione dell'organizzazione.

- Crea SCPs. Per ulteriori informazioni sulle autorizzazioni minime, consultate Creazione di un SCP.
- Allega l'SCP agli account o alle unità organizzative di destinazione (). OUs Per ulteriori informazioni sulle autorizzazioni minime, vedere Allegare e scollegare le politiche di controllo del servizio.

Limitazioni

- SCPs non influiscono sugli utenti o sui ruoli nell'account di gestione. Influiscono solo sugli account membri nell'organizzazione.
- SCPs riguardano solo gli utenti e i ruoli AWS Identity and Access Management (IAM) gestiti da account che fanno parte dell'organizzazione. Per ulteriori informazioni, consulta Effetti delle SCP sulle autorizzazioni.

Tools (Strumenti)

Servizi AWS

- AWS Organizationsè un servizio di gestione degli account che consente di consolidare più account Account AWS in un'organizzazione da creare e gestire centralmente. In questo modello, si utilizzano le politiche di controllo del servizio (SCPs) in AWS Organizations.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.

Best practice

Dopo aver stabilito questo SCP nella tua organizzazione, assicurati di aggiornarlo frequentemente per risolvere eventuali novità Servizi AWS o funzionalità che potrebbero influire sull'accesso a Internet.

Epiche

Crea e collega l'SCP

Operazione	Description	Competenze richieste
Crea l'SCP.	1. Accedi alla console AWS Organizations. È necessario accedere all'account di gestione dell'organizzazione. 2. Nel riquadro di sinistra, scegli Politiche. 3. Nella pagina delle politiche, scegli Criteri di controllo del servizio. 4. Nella pagina Service control policies (Policy di controllo dei servizi), scegli Create policy (Crea policy). 5. Nella pagina Crea una nuova politica di controllo del servizio, inserisci il nome della politica e una descrizione facoltativa della politica. 6. (Facoltativo) Aggiungi AWS tag alla tua politica. 7. Nell'editor JSON, elimina la politica dei segnaposto. 8. Incollare la seguente policy nell'editor JSON. `{ "Version": "2012-10-17",`	Amministratore AWS

Operazione	Description	Competenze richieste

	<pre>{
	  "Statement": [
	    {
	      "Action": [
	        "ec2:Atta
	chInternetGateway",
	        "ec2:Crea
	teInternetGateway",
	
	        "ec2:Crea
	teVpcPeeringConnec
	tion",
	        "ec2:Acce
	ptVpcPeeringConnec
	tion",
	        "ec2:Crea
	teEgressOnlyIntern
	etGateway"
	      ],
	      "Resource":
	 "*",
	      "Effect": "Deny"
	    },
	    {
	      "Action": [
	        "globalac
	celerator:Create*",
	        "globalac
	celerator:Update*"
	      ],
	      "Resource":
	 "*",
	      "Effect": "Deny"
	    }
	  ]
	}</pre>
	9. Scegli Crea policy.	

Operazione	Description	Competenze richieste
Collega l'SCP.	1. Nella pagina delle politiche di controllo del servizio, scegli la politica che hai creato. 2. Nella scheda Targets (Destinazioni), scegli Attach (Collega). 3. Seleziona l'unità organizzativa o l'account a cui desideri allegare la politica. Potrebbe essere necessario espanderla OUs per trovare l'unità organizzativa o l'account desiderato. 4. Scegli Collega policy.	Amministratore AWS

Risorse correlate

- AWS Organizations documentazione
- politiche di controllo del servizio (SCPs)
- Architettura di ispezione centralizzata con AWS Gateway Load Balancer AWS Transit Gateway e AWS (post sul blog)

Esporta un report delle identità di AWS IAM Identity Center e delle relative assegnazioni utilizzando PowerShell

Jorge Pava, Frank Allotta, Manideep Reddy Gillela e Ciad Miles, Amazon Web Services

Riepilogo

Quando utilizzi AWS IAM Identity Center (successore di AWS Single Sign-On) per gestire centralmente l'accesso Single Sign-On (SSO) a tutti gli account e le applicazioni cloud di Amazon Web Services (AWS), la segnalazione e il controllo di tali assegnazioni tramite la Console di gestione AWS possono essere noiosi e richiedere molto tempo. Ciò è particolarmente vero se stai segnalando le autorizzazioni per un utente o un gruppo su dozzine o centinaia di account AWS.

Per molti, lo strumento ideale per visualizzare queste informazioni sarebbe utilizzare un'applicazione per fogli di calcolo, come Microsoft Excel. Questo può aiutarti a filtrare, cercare e visualizzare i dati per l'intera organizzazione, gestita da AWS Organizations.

Questo modello descrive come utilizzare AWS Tools per PowerShell generare un report sulle configurazioni di identità SSO in IAM Identity Center. Il report è formattato come file CSV e include il nome dell'identità (principale), il tipo di identità (utente o gruppo), gli account a cui l'identità può accedere e i set di autorizzazioni. Dopo aver generato questo rapporto, puoi aprirlo nell'applicazione preferita per cercare, filtrare e controllare i dati secondo necessità. L'immagine seguente mostra dati di esempio in un'applicazione per fogli di calcolo.

Importante: poiché questo rapporto contiene informazioni riservate, consigliamo vivamente di archiviarle in modo sicuro e condividerle solo su base individuale. need-to-know

Prerequisiti e limitazioni

Prerequisiti

- IAM Identity Center e AWS Organizations, configurati e abilitati.
- PowerShell, installato e configurato. Per ulteriori informazioni, vedere Installazione PowerShell (documentazione Microsoft).
- Strumenti AWS per PowerShell, installati e configurati. Per motivi di prestazioni, consigliamo vivamente di installare la versione modulare di AWS Tools for PowerShell, chiamata.

`AWS.Tools` Ogni servizio AWS è supportato da un piccolo modulo individuale. Nella PowerShell shell, inserisci i seguenti comandi per installare i moduli necessari per questo modello: `AWS.Tools.InstallerOrganizations`,`SSOAdmin`, `eIdentityStore`.

```
Install-Module AWS.Tools.Installer
Install-AWSToolsModule -Name Organizations, SSOAdmin, IdentityStore
```

Per ulteriori informazioni, consulta Installare AWS.Tools su Windows o Installare AWS.Tools su Linux o macOS (AWS Tools per la documentazione). PowerShell Se ricevi un errore durante l'installazione dei moduli, consulta la sezione Risoluzione dei problemi di questo schema.

- AWS Command Line Interface (AWS CLI) o l'SDK AWS devono essere precedentemente configurati con credenziali di lavoro effettuando una delle seguenti operazioni:
  - Usa l'interfaccia a riga di comando di AWS `aws configure` Per ulteriori informazioni, consulta Quick configuration (documentazione AWS CLI).
  - Configura AWS CLI o AWS Cloud Development Kit (AWS CDK) per ottenere l'accesso temporaneo tramite un ruolo AWS Identity and Access Management (IAM). Per ulteriori informazioni, consulta Ottenere le credenziali del ruolo IAM per l'accesso alla CLI (documentazione IAM Identity Center).
- Un profilo denominato per l'AWS CLI che ha salvato le credenziali per un principal IAM che:
  - Ha accesso all'account di gestione AWS Organizations o all'account amministratore delegato per IAM Identity Center
  - Le politiche gestite da `AWSSSODirectoryReadOnly` AWS `AWSSSOReadOnly` e AWS sono state applicate ad esso?

  Per ulteriori informazioni, consulta Using named profiles (documentazione AWS CLI) e AWS managed policy (documentazione IAM).

Limitazioni

- Gli account AWS di destinazione devono essere gestiti come organizzazione in AWS Organizations.

Versioni del prodotto

- Per tutti i sistemi operativi, si consiglia di utilizzare la PowerShell versione 7.0 o successiva.

Architecture

Architettura Target

1. L'utente esegue lo script in una PowerShell riga di comando.
2. Lo script presuppone il profilo denominato per AWS CLI. Ciò consente l'accesso a IAM Identity Center.
3. Lo script recupera le configurazioni di identità SSO da IAM Identity Center.
4. Lo script genera un file CSV nella stessa directory sulla workstation locale in cui viene salvato lo script.

Tools (Strumenti)

Servizi AWS

- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.
- AWS IAM Identity Center ti aiuta a gestire centralmente l'accesso Single Sign-On (SSO) a tutti i tuoi account AWS e applicazioni cloud.
- AWS Tools for PowerShell è un set di PowerShell moduli che ti aiutano a creare script di operazioni sulle tue risorse AWS dalla PowerShell riga di comando.

Altri strumenti

- PowerShellè un programma di gestione dell'automazione e della configurazione di Microsoft che funziona su Windows, Linux e macOS.

Epiche

Genera il rapporto

Operazione	Description	Competenze richieste
Prepara la sceneggiatura.	1. Copia lo PowerShell script nella sezione Informazi	Amministratore cloud

Operazione	Description	Competenze richieste
	oni aggiuntive di questo modello. 2. Nella `Param` sezione, per il tuo ambiente AWS, definisci i valori per le seguenti variabili: • `OutputFile` — Il nome del file del report. • `ProfileName` — Il profilo denominato della CLI AWS che desideri utilizzare per generare il report. • `Region`— La regione AWS in cui è distribui to IAM Identity Center. Per un elenco completo delle regioni e dei relativi codici, consulta Endpoint regionali. 3. Salva lo script con il nome `SSO-Report.ps1` del file.	

Operazione	Description	Competenze richieste

Eseguire lo script.	Si consiglia di eseguire lo script personalizzato nella PowerShell shell con il seguente comando.
	`.\SSO-Report.ps1`
	In alternativa, è possibile eseguire lo script da un'altra shell immettendo il seguente comando.
	`pwsh .\SSO-Report.ps1`
	Lo script genera un file CSV nella stessa directory del file di script.
		Amministratore cloud

Analizza i dati del report.	Il file CSV di output contiene le intestazioni AccountNa mePermissionSet, Principal e Type. Apri questo file nell'appl icazione per fogli di calcolo preferita. È possibile creare una tabella di dati per filtrare e ordinare l'output.	Amministratore cloud

risoluzione dei problemi

Problema	Soluzione
Errore `The term 'Get-<parameter>' is not recognized as the name of`	AWS Tools for PowerShell o i relativi moduli non sono installati. Nella PowerShell shell, inserisci i seguenti comandi per installare AWS

Problema	Soluzione

`a cmdlet, function, script file, or operable program.`	Tools for PowerShell e i moduli necessari per questo modello: `AWS.Tools.Installer` `Organizations` ,`SSOAdmin`, `eIdentityStore` .
	
	`Install-Module AWS.Tools.Installer`
	`Install-AWSToolsModule -Name Organizations, SSOAdmin, IdentityStore`

Errore `No credentials specified or obtained from persisted/shell defaults`	Nella sezione Prepara lo script nella sezione Epics, conferma di aver inserito correttamente `Region` le variabili `ProfileName` and. Assicurati che le impostazioni e le credenziali nel profilo indicato dispongano di autorizzazioni sufficienti per amministrare IAM Identity Center.
`Authenticode Issuer …` errore durante l'installazione dei moduli AWS.Tools	Aggiungi il `-SkipPublisherCheck` parametro alla fine del `Install-AWSToolsModule` comando.

Problema	Soluzione

Errore `Get-ORGAccountList : Assembly AWSSDK.SSO could not be found or loaded.`	Questo errore può verificarsi quando vengono specificati profili AWS CLI denominati, AWS CLI è configurato per autenticare gli utenti con IAM Identity Center e AWS CLI è configura to per recuperare automaticamente i token di autenticazione aggiornati. Per risolvere questo errore, procedi come segue:
	1. Immettete il comando seguente per confermare che i `SSOOIDC` moduli `SSO` and sono installati.
	`Install-AWSToolsModule SSO, SSOOIDC`
	2. Inserisci le seguenti righe nello script sotto il `param()` blocco.
	`Import-Module AWS.Tools.SSO`
	`Import-Module AWS.Tools.SSOOIDC`

Risorse correlate

- Dove vengono memorizzate le impostazioni di configurazione? (documentazione dell'interfaccia a riga di comando AWS)
- Configurazione dell'interfaccia a riga di comando di AWS per l'utilizzo di AWS IAM Identity Center (documentazione AWS CLI)
- Utilizzo di profili denominati (documentazione AWS CLI)

Informazioni aggiuntive

Nel seguente script, determina se è necessario aggiornare i valori per i seguenti parametri:

- Se utilizzi un profilo denominato nell'interfaccia a riga di comando di AWS per accedere all'account in cui è configurato IAM Identity Center, aggiorna il `$ProfileName` valore.
- Se IAM Identity Center è distribuito in una regione AWS diversa dalla regione predefinita per la tua configurazione AWS CLI o AWS SDK, aggiorna `$Region` il valore per utilizzare la regione in cui è distribuito IAM Identity Center.
- Se nessuna di queste situazioni si applica, non è richiesto alcun aggiornamento dello script.

```
param (
    # The name of the output CSV file
    [String] $OutputFile  = "SSO-Assignments.csv",
    # The AWS CLI named profile
    [String] $ProfileName = "",
    # The AWS Region in which IAM Identity Center is configured
    [String] $Region      = ""
)
$Start = Get-Date; $OrgParams = @{}
If ($Region){ $OrgParams.Region = $Region}
if ($ProfileName){$OrgParams.ProfileName = $ProfileName}
$SSOParams   = $OrgParams.Clone(); $IdsParams = $OrgParams.Clone()
$AccountList = Get-ORGAccountList @OrgParams | Select-Object Id, Name
$SSOinstance = Get-SSOADMNInstanceList @OrgParams
$SSOParams['InstanceArn']       = $SSOinstance.InstanceArn
$IdsParams['IdentityStoreId']   = $SSOinstance.IdentityStoreId
$PSsets       = @{}; $Principals   = @{}
$Assignments  = @(); $AccountCount = 1; Write-Host ""
foreach ($Account in $AccountList) {
    $Duration = New-Timespan -Start $Start -End (Get-Date) | ForEach-Object
 {[Timespan]::New($_.Days, $_.Hours, $_.Minutes, $_.Seconds)}
    Write-Host "`r$Duration - Account $AccountCount of $($AccountList.Count)
 (Assignments:$($Assignments.Count))        " -NoNewline
    $AccountCount++
    foreach ($PS in Get-SSOADMNPermissionSetsProvisionedToAccountList -AccountId
 $Account.Id @SSOParams) {
        if (-not $PSsets[$PS]) {$PSsets[$PS] = (Get-SSOADMNPermissionSet @SSOParams -
PermissionSetArn $PS).Name;$APICalls++}
        $AssignmentsResponse = Get-SSOADMNAccountAssignmentList @SSOParams -
PermissionSetArn $PS -AccountId $Account.Id
        if ($AssignmentsResponse.NextToken) {$AccountAssignments =
 $AssignmentsResponse.AccountAssignments}
        else {$AccountAssignments = $AssignmentsResponse}
        While ($AssignmentsResponse.NextToken) {
```

```
            $AssignmentsResponse = Get-SSOADMNAccountAssignmentList @SSOParams -
PermissionSetArn $PS -AccountId $Account.Id -NextToken $AssignmentsResponse.NextToken
            $AccountAssignments += $AssignmentsResponse.AccountAssignments}
        foreach ($Assignment in $AccountAssignments) {
            if (-not $Principals[$Assignment.PrincipalId]) {
                $AssignmentType = $Assignment.PrincipalType.Value
                $Expression     = "Get-IDS"+$AssignmentType+" @IdsParams -"+
$AssignmentType+"Id "+$Assignment.PrincipalId
                $Principal      = Invoke-Expression $Expression
                if ($Assignment.PrincipalType.Value -eq "GROUP")
 { $Principals[$Assignment.PrincipalId] = $Principal.DisplayName }
                else { $Principals[$Assignment.PrincipalId] = $Principal.UserName }
            }
            $Assignments += [PSCustomObject]@{
                AccountName     = $Account.Name
                PermissionSet   = $PSsets[$PS]
                Principal       = $Principals[$Assignment.PrincipalId]
                Type            = $Assignment.PrincipalType.Value}
        }
    }
}
$Duration = New-Timespan -Start $Start -End (Get-Date) | ForEach-Object
 {[Timespan]::New($_.Days, $_.Hours, $_.Minutes, $_.Seconds)}
Write-Host "`r$($AccountList.Count) accounts done in $Duration. Outputting result to
 $OutputFile"
$Assignments | Sort-Object Account | Export-CSV -Path $OutputFile -Force
```

Limita l'accesso in base all'indirizzo IP o alla geolocalizzazione utilizzando AWS WAF

Louis Hourcade, Amazon Web Services

Riepilogo

AWS WAFè un firewall per applicazioni Web che aiuta a proteggere le applicazioni Web e APIs da exploit e bot Web comuni che possono influire sulla disponibilità, compromettere la sicurezza o consumare risorse eccessive. Gli elenchi di controllo degli accessi Web (Web ACLs) AWS WAF consentono di controllare il modo in cui il traffico raggiunge le applicazioni. In un ACL Web, aggiungi regole o gruppi di regole progettati per consentire il traffico legittimo, controllare il traffico dei bot e bloccare i modelli di attacco comuni. Per ulteriori informazioni, consulta How AWS WAF works.

Puoi associare i seguenti tipi di regole al tuo AWS WAF Web ACLs:

- Gruppi di regole AWS gestiti: i team e Marketplace AWS i venditori di Managed Rules offrono set di regole preconfigurati. Alcuni gruppi di regole gestite sono progettati per aiutare a proteggere tipi specifici di applicazioni Web. Altri offrono un'ampia protezione contro minacce note o vulnerabilità comuni.
- Regole personalizzate e gruppi di regole personalizzati: puoi anche creare regole e gruppi di regole che personalizzano l'accesso alle tue applicazioni Web e APIs. Ad esempio, puoi limitare il traffico in base a un elenco specifico di indirizzi IP o a un elenco di paesi.

Utilizzando questo modello e l'archivio di codice associato, è possibile utilizzare il AWS Cloud Development Kit (AWS CDK)per distribuire il AWS WAF Web ACLs con regole personalizzate. Queste regole limitano l'accesso alle risorse delle applicazioni Web in base all'indirizzo IP o alla geolocalizzazione dell'utente finale. Facoltativamente, puoi anche collegare diversi gruppi di regole gestite.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Autorizzazioni per distribuire risorse AWS WAF
- AWS CDK, installato e configurato nel tuo account

- Git, installato

Limitazioni

- È possibile utilizzare questo modello solo Regioni AWS dove AWS WAF è disponibile. Per la disponibilità per regione, vedi Servizi AWS per regione.

Tools (Strumenti)

Servizi AWS

- AWS Cloud Development Kit (AWS CDK)è un framework di sviluppo software che consente di definire e fornire Cloud AWS l'infrastruttura in codice.
- AWS WAFè un firewall per applicazioni Web che consente di monitorare le richieste HTTP e HTTPS inoltrate alle risorse protette delle applicazioni Web.

Archivio di codice

Il codice per questo pattern è disponibile nella restrizione GitHub IP e geolocalizzazione con repository. AWS WAF Il codice distribuisce due web. AWS WAF ACLs Il primo è un ACL web regionale destinato alle risorse di Amazon API Gateway. Il secondo è l'ACL web globale per CloudFront le risorse Amazon. Entrambi i siti Web ACLs contengono le seguenti regole personalizzate:

- `IPMatch`blocca le richieste provenienti da indirizzi IP non consentiti.
- `GeoMatch`blocca le richieste provenienti da paesi non consentiti.

Durante la distribuzione, puoi facoltativamente collegare tutti i seguenti gruppi di regole gestite al tuo web: ACLs

- Core rule set (CRS): questo gruppo di regole contiene regole generalmente applicabili alle applicazioni Web. Aiuta a proteggere dallo sfruttamento di un'ampia gamma di vulnerabilità, incluse alcune delle vulnerabilità ad alto rischio e più comuni descritte nelle pubblicazioni OWASP, come OWASP Top 10.
- Protezione degli amministratori: questo gruppo di regole contiene regole che consentono di bloccare l'accesso esterno alle pagine amministrative esposte.

- Input noti non validi: questo gruppo di regole aiuta a bloccare i modelli di richiesta noti per non essere validi e associati allo sfruttamento o all'individuazione di vulnerabilità.
- Elenco di reputazione IP di Amazon: questo gruppo di regole contiene regole basate sull'intelligence interna delle minacce di Amazon. Ti aiuta a bloccare gli indirizzi IP che in genere sono associati a bot o altre minacce.
- Gruppo di regole gestito dal sistema operativo Linux: questo gruppo di regole aiuta a bloccare i modelli di richiesta associati allo sfruttamento delle vulnerabilità di Linux, inclusi gli attacchi LFI (Local File Inclusion) specifici di Linux.
- Gruppo di regole gestito dal database SQL: questo gruppo di regole aiuta a bloccare i modelli di richiesta associati allo sfruttamento dei database SQL, come gli attacchi SQL injection.

Epiche

Configura il web AWS WAF ACLs

Operazione	Description	Competenze richieste
Clonare il repository.	Immettete il seguente comando per clonare l'IP e la restrizione di geolocalizzazione con AWS WAF repository sulla workstation locale: `git clone https://github.com/aws-samples/ip-and-geolocation-restriction-with-waf-cdk.git`	Git
Configura le regole.	1. Nel repository clonato, apri il file app.py. 2. Modificate i valori delle seguenti variabili per personalizzare le regole:	Informazioni generali su AWS, Python

Operazione	Description	Competenze richieste

	<pre>aws_acccount =
	"AWS_ACCOUNT"
	region = "AWS_REGI
	ON"
	ip_list = ["CIDR_RA
	NGE_1", "CIDR_RAN
	GE_2"]
	geo_list = ["COUNTRY
	_CODE_1", "COUNTRY_
	CODE_2"]
	aws_managed_rules =
	True</pre> Dove:
	• `aws_account` è l'ID della destinazione Account AWS.
	• `region`è la destinazione dell'ACL Web Regione AWS per le risorse API Gateway.
	Note
	L'ACL web per CloudFront le risorse è globale e verrà distribui to nella regione. `us-east-1`
	• `ip_list`è l'elenco degli intervalli CIDR a cui è consentito l'accesso.	

Operazione	Description	Competenze richieste
	• `geo_list` è l'elenco dei paesi a cui è consentito l'accesso. Per i valori validi, consulta la AWS WAF documentazione. • `aws_managed_rules` controlla se i gruppi di regole gestiti vengono aggiunti all'ACL Web. Se questo valore è `True`, vengono aggiunti. Se questo valore è `False`, vengono esclusi. 3. Salvate e chiudete il file app.py.	

Esegui il bootstrap e distribuisci il codice

Operazione	Description	Competenze richieste
Avvia il tuo AWS ambiente.	Se non l'hai già fatto, devi avviare l' AWS ambiente prima di poter distribuire l'applicazione. AWS CDK 1. Nella AWS CDK CLI, inserisci il seguente comando per avviare la regione: `us-east-1` `cdk bootstrap aws://<account-id>/us-east-1`	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	2. Se stai distribuendo l'ACL web per API Gateway in una regione diversa da `us-east-1` , inserisci il seguente comando per avviare la regione di destinazione: `cdk bootstrap aws://<account-id>/<region>`	
Implementa l' AWS CDK applicazione.	1. Immettete il seguente comando per distribuire l' AWS CDK applicazione: `cdk deploy --all` 2. Attendi il completamento della distribuzione AWS CloudFormation dello stack.	Informazioni generali su AWS

Convalida la distribuzione

Operazione	Description	Competenze richieste
Conferma che il Web sia ACLs stato distribuito correttamente.	1. Accedi a Console di gestione AWS, quindi apri la AWS WAF console. 2. Nel riquadro di navigazione, scegli Web ACLs. 3. Nell'elenco di Regioni AWS, scegli Global (CloudFront).	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	4. Verifica che il nuovo ACL CloudFront web sia stato distribuito e conferma che abbia l'indirizzo IP e le regole di geolocalizzazione che hai definito. Il nome predefinito di questo ACL web è. `WebACLCloudfront-<ID>` 5. Nell'elenco di Regioni AWS, scegli la regione in cui hai distribuito lo stack. 6. Verifica che sia stato distribuito un nuovo ACL Web per le risorse API Gateway. Verifica che disponga dell'indirizzo IP e delle regole di geolocalizzazione che hai definito. Il nome predefinito di questo ACL web è. `WebACLApiGW-<ID>`	

Operazione	Description	Competenze richieste
(Facoltativo) Associa il Web ACLs alle tue risorse.	Associate il AWS WAF Web ACLs alle vostre AWS risorse, come un Application Load Balancer, un API Gateway o CloudFront una distribuz ione. Per istruzioni, consulta Associare o dissociare un ACL Web a una risorsa.AWS Per un esempio, vedete class CfnWeb ACLAssociation (construct) nella documenta zione. AWS CDK	Informazioni generali su AWS

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Eliminare le pile.	1. Dissocia l'ACL Web da qualsiasi risorsa. AWS Per istruzioni, consultate la documentazione.AWS WAF 2. Nella AWS CDK CLI, immettete il seguente comando per eliminare l' AWS CDK applicazione. `cdk destroy --all`	Informazioni generali su AWS

Risorse correlate

- Riferimento alle API (AWS CDK documentazione)
- aws-cdk-libmodulo.aws_wafv2 (documentazione)AWS CDK

- Lavorare con il web (ACLsdocumentazione AWS WAF)
- Gestire i propri gruppi di regole (AWS WAF documentazione)
- Regole (AWS WAF documentazione)

Scansiona gli archivi Git alla ricerca di informazioni sensibili e problemi di sicurezza utilizzando git-secrets

Saurabh Singh, Amazon Web Services

Riepilogo

Questo modello descrive come utilizzare lo strumento open source git-secrets di AWS Labs per scansionare i repository di sorgenti Git e trovare codice che potrebbe potenzialmente includere informazioni sensibili, come password utente o chiavi di accesso, AWS o che presenta altri problemi di sicurezza.

`git-secrets`analizza i commit, i messaggi di commit e le unioni per impedire che informazioni sensibili come quelle segrete vengano aggiunte ai tuoi repository Git. Ad esempio, se un commit, un messaggio di commit o qualsiasi commit in una cronologia di unione corrisponde a uno dei modelli di espressioni regolari proibiti e configurati, il commit viene rifiutato.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Un repository Git che richiede una scansione di sicurezza
- Un client Git (versione 2.37.1 e successive) installato

Architecture

Architettura Target

- Git
- `git-secrets`

Tools (Strumenti)

- git-secrets è uno strumento che ti impedisce di inserire informazioni sensibili nei repository Git.

- Git è un sistema di controllo delle versioni distribuito open source.

Best practice

- Scansiona sempre un repository Git includendo tutte le revisioni:

```
git secrets --scan-history
```

Epiche

Connect a un' EC2 istanza

Operazione	Description	Competenze richieste
Connect a un' EC2 istanza utilizzando SSH.	Connettiti a un'istanza Amazon Elastic Compute Cloud (Amazon EC2) utilizzando SSH e un file key pair. Puoi saltare questo passaggio se stai scansionando un repository sul tuo computer locale.	Informazioni generali su AWS

Installa Git

Operazione	Description	Competenze richieste
Installa Git.	Installa Git usando il comando: `yum install git -y` Se stai usando il tuo computer locale, puoi installare un client Git per una versione specifica del sistema operativo. Per	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	ulteriori informazioni, consulta il sito Web Git.	

Clona il repository dei sorgenti e installa git-secrets

Operazione	Description	Competenze richieste
Clona il repository dei sorgenti Git.	Per clonare il repository Git che vuoi scansionare, scegli il comando Git clone dalla tua home directory.	Informazioni generali su AWS
Clona git-secrets.	Clona il repository `git-secrets` Git. `git clone https://github.com/awslabs/git-secrets.git` Posizionalo `git-secrets` da qualche parte nel tuo in `PATH` modo che Git lo raccolga quando corri`git-secrets` .	Informazioni generali su AWS
Installa git-secrets.	Per Unix e varianti (Linux/macOS): È possibile utilizzare la `install` destinazione di `Makefile` (fornita nel `git-secrets` repository) per installare lo strumento. È possibile personalizzare il percorso di installazione	Informazioni generali su AWS

Operazione	Description	Competenze richieste

	utilizzando le `MANPREFIX` variabili `PREFIX` and.
	
	`make install`
	
	Per Windows:
	
	Esegui lo PowerShell `install.ps1` script fornito nel `git-secrets` repository. Questo script copia i file di installazione in una directory di installazione (`%USERPROFILE%/.git-secrets` per impostazione predefinita) e aggiunge la directory all'utente `PATH` corrente.
	
	`PS > ./install.ps1`
	
	Per Homebrew (utenti macOS):
	
	Esegui:
	
	`brew install git-secrets`	

Scansiona l'archivio di codice git

Operazione	Description	Competenze richieste
Vai al repository dei sorgenti.	Passa alla directory del repository Git che desideri scansionare:	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	`cd my-git-repository`	
Registra il set di regole AWS (Git hooks).	`git-secrets` Per configurare la scansione del tuo repository Git su ogni commit, esegui il comando: `git secrets --register-aws`	Informazioni generali su AWS
Scansiona il repository.	Esegui il seguente comando per avviare la scansione del repository: `git secrets --scan`	Informazioni generali su AWS

Operazione	Description	Competenze richieste

Esamina il file di output.	Lo strumento genera un file di output se rileva una vulnerabilità nel tuo repository Git. Esempio:
	
	`example.sh:4:AWS_SECRET_ACCESS_KEY = *********`
	
	`[ERROR] Matched one or more prohibited patterns`
	
	`Possible mitigations:`
	`- Mark false positives as allowed using: git config --add secrets.allowed ...`
	`- Mark false positives as allowed by adding regular expressions to .gitallowed at repository's root directory`
	`- List your configured patterns: git config --get-all secrets.patterns`
	`- List your configured allowed patterns: git config --get-all secrets.allowed`
	`- List your configured allowed patterns in .gitallowed at repository's root directory`
		Informazioni generali su AWS

Operazione	Description	Competenze richieste
	```- Use --no-verify if this is a one-time false positive```	

Risorse correlate

- strumento git-secrets

Trasferimenti sicuri di file utilizzando Transfer Family, Amazon Cognito e GuardDuty

Manoj Kumar, Amazon Web Services

Riepilogo

Questa soluzione consente di trasferire file in modo sicuro tramite un server SFTP utilizzando. AWS Transfer Family Include funzionalità di scansione antimalware automatizzata tramite Malware Protection for S3, una funzionalità di Amazon GuardDuty. È progettato per le organizzazioni che devono scambiare file in modo sicuro con parti esterne e verificare che tutti i file in arrivo vengano scansionati alla ricerca di malware prima di essere elaborati.

I modelli Infrastructure as Code (IaC) forniti con questo modello consentono di implementare quanto segue:

- Un server SFTP sicuro con autenticazione Amazon Cognito tramite AWS Lambda
- Bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) per i caricamenti e i file in entrata sottoposti a scansione alla ricerca di malware
- Un'architettura basata su cloud privato virtuale (VPC) con sottoreti pubbliche e private su più zone di disponibilità
- Controllo degli accessi basato su IP per il traffico in ingresso e in uscita, con elenchi di autorizzazioni e negazioni configurabili
- Scansione antimalware automatizzata tramite GuardDuty
- Routing intelligente dei file basato sui risultati della scansione tramite Amazon EventBridge e Lambda
- Notifiche in tempo reale per incidenti di sicurezza tramite Amazon Simple Notification Service (Amazon SNS)
- Crittografia per bucket Amazon S3 e variabili di ambiente Lambda tramite () AWS Key Management Service AWS KMS
- Endpoint Amazon Virtual Private Cloud (Amazon VPC) per l'accesso senza esposizione a Internet
- Registrazione completa tramite l'integrazione con Amazon CloudWatch

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Autorizzazioni in AWS Identity and Access Management (IAM) per eseguire le azioni descritte in questo modello, inclusa la distribuzione di AWS CloudFormation modelli che forniscono ruoli IAM
- GuardDuty, abilitato nell'account di destinazione
- Protezione da malware per S3, abilitata nell'account di destinazione
- Le quote di servizio consentono di creare quanto segue nell'account di destinazione:
  - Un VPC
  - Una sottorete privata
  - Una sottorete pubblica
  - Tre indirizzi IP elastici
  - Limiti di concorrenza Lambda sufficienti
- Un indirizzo e-mail valido per le notifiche relative alla sicurezza
- (Facoltativo) Un elenco di indirizzi IP o intervalli CIDR che desideri consentire o negare
- (Facoltativo) AWS Command Line Interface (AWS CLI), installato e configurato

Limitazioni

- La protezione da malware per S3 è soggetta a quote, come la dimensione massima dei file. Per ulteriori informazioni, consulta Quotas in Malware Protection for S3 e Supportabilità delle funzionalità di Amazon S3 nella documentazione. GuardDuty
- Questa soluzione utilizza solo l'autenticazione con nome utente e password di Amazon Cognito. I metodi di autenticazione basati su certificati o altri metodi di autenticazione non sono supportati in questo modello. Per impostazione predefinita, questa soluzione non configura l'autenticazione a più fattori (MFA).
- La soluzione implementa il controllo degli accessi basato su IP solo tramite gruppi di sicurezza.

Architecture

Il seguente diagramma di architettura mostra le risorse distribuite secondo questo schema. Questa soluzione utilizza Amazon Cognito per l'autenticazione e l'autorizzazione degli utenti. Per il

caricamento dei file viene utilizzato un server AWS Transfer Family SFTP. I file vengono archiviati in bucket Amazon S3 e Amazon GuardDuty esegue la scansione dei file alla ricerca di malware. Amazon SNS invia una notifica e-mail se viene rilevato un malware.

Il diagramma mostra il flusso di lavoro seguente:

1. Un utente si connette all'endpoint del server SFTP in. AWS Transfer Family Questo avvia il processo di autenticazione con il pool di utenti di Amazon Cognito.
2. Una funzione Lambda avvia il processo di autenticazione e autorizzazione e convalida le credenziali dell'utente con Amazon Cognito.
3. La funzione Lambda restituisce il `UploadBucket` bucket Amazon S3 come directory home. L'utente assume il ruolo IAM per il server Transfer Family e la funzione Lambda notifica all'utente che è stato autenticato correttamente.
4. L'utente carica un file sul server SFTP Transfer Family. Il file viene archiviato nel `UploadBucket` bucket Amazon S3.
5. GuardDuty esegue la scansione del file alla ricerca di malware. I potenziali risultati della scansione sono`NO_THREATS_FOUND`,`THREATS_FOUND`, `UNSUPPORTEDACCESS_DENIED`, e`FAILED`. Per risultati di esempio, consulta il risultato della scansione degli oggetti S3 nella GuardDuty documentazione.
6. Una EventBridge regola rileva l'evento del risultato della scansione.
7. EventBridge avvia la funzione Lambda per il routing dei file.
8. La funzione Lambda elabora l'evento e filtra i file in base ai risultati della scansione come segue:
   - I file con un risultato di `NO_THREATS_FOUND` scansione vengono inviati al bucket `CleanBucket` Amazon S3.
   - I file con un risultato di `THREATS_FOUND` scansione vengono inviati al bucket `MalwareBucket` Amazon S3.
   - I file con un risultato di `UNSUPPORTED` scansione vengono inviati al bucket `ErrorBucket` Amazon S3.
   - I file con un risultato di `ACCESS_DENIED` scansione vengono inviati al bucket `ErrorBucket` Amazon S3.
   - I file con un risultato di `FAILED` scansione vengono inviati al bucket `ErrorBucket` Amazon S3.

   Tutti i file sono crittografati con un. AWS KMS key

9. Se un file è stato inviato al bucket `MalwareBucket` Amazon S3, la funzione Lambda avvia un argomento Amazon SNS. L'argomento Amazon SNS invia una notifica e-mail a un indirizzo e-mail che configuri.

Tools (Strumenti)

Servizi AWS

- Amazon ti CloudWatch aiuta a monitorare i parametri delle tue AWS risorse e delle applicazioni su cui esegui AWS in tempo reale.
- Amazon Cognito fornisce autenticazione, autorizzazione e gestione degli utenti per app Web e mobili.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altro modo. Account AWS
- Amazon GuardDuty è un servizio di monitoraggio continuo della sicurezza che analizza ed elabora i log per identificare attività impreviste e potenzialmente non autorizzate nel tuo ambiente. AWS
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS Transfer Familyti aiuta a trasferire file da e verso i servizi di AWS storage tramite i protocolli SFTP, FTPS o FTP.
- Amazon Virtual Private Cloud (Amazon VPC) ti aiuta a lanciare AWS risorse in una rete virtuale che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.

Repository di codice

Il codice di questo pattern è disponibile nel repository GitHub AWS Transfer Family and GuardDuty Malware Scanning Solution.

Best practice

Il CloudFormation modello fornito è progettato per incorporare molte AWS best practice, come le autorizzazioni con privilegi minimi per i ruoli e le policy IAM, la crittografia a riposo e in transito e la rotazione automatica delle chiavi. Per gli ambienti di produzione, valuta la possibilità di implementare i seguenti consigli aggiuntivi:

- Abilita l'autenticazione a più fattori per gli utenti di Amazon Cognito
- Implementa AWS Shieldla protezione da denial of service (S) DDo distribuita
- Configurazione AWS Configper il monitoraggio continuo della conformità
- AWS CloudTrailImplementa per una registrazione completa delle API
- Configura Amazon GuardDuty per il rilevamento delle minacce oltre alla scansione dei malware
- Implementa AWS Security Hub CSPMper la gestione centralizzata della sicurezza
- Utilizzalo Gestione dei segreti AWSper la gestione delle credenziali
- Implementa il monitoraggio del traffico di rete con Traffic Mirroring
- Configura Amazon Macie per il rilevamento e la protezione dei dati sensibili in Amazon S3
- Implementa valutazioni di sicurezza e test di penetrazione regolari
- Stabilisci un piano formale di risposta agli incidenti
- Implementa l'applicazione di patch automatizzate per tutti i componenti
- Conduci corsi di formazione periodici sulla sicurezza per gli amministratori
- Configurazione AWS Organizationsper la gestione della sicurezza su più account

Epiche

Implementa le risorse

Operazione	Description	Competenze richieste
Clonare il repository.	Immettete il seguente comando per clonare l'AWS Transfer Family archivio della soluzione di scansione	Sviluppatore di app, ingegnere DevOps

Operazione	Description	Competenze richieste
	GuardDuty antimalware sulla workstation locale: `git clone https://github.com/aws-samples/sample-secure-transfer-family-code.git`	

Operazione	Description	Competenze richieste
Crea lo CloudFormation stack.	1. Segui le istruzioni nella CloudFormation documentazione per creare uno stack tramite la CloudFormation console o tramite. AWS CLI Configura i seguenti parametri: • `UploadBucketName` - Carica il nome del bucket • `CleanBucketName` - Pulisci il nome del bucket dei file • `MalwareBucketName` - Nome del bucket dei file di malware • `ErrorBucketName` - Nome del bucket dei file di errore • `SecurityTeamEmail` - Indirizzo e-mail per gli avvisi • `VpcCIDR`- Gamma CIDR per VPC • `PublicSubnet1CIDR` - Intervallo CIDR per la sottorete pubblica 1 • `PublicSubnet2CIDR` - Intervallo CIDR per la sottorete pubblica 2 • `PrivateSubnet1CIDR` - Intervallo CIDR per sottorete privata 1	Amministratore del cloud, ingegnere DevOps

Operazione	Description	Competenze richieste
	• `PrivateSubnet2CIDR` - Intervallo CIDR per sottorete privata 2 • `AllowedIPAddresses` - Indirizzi IP o intervalli CIDR consentiti • `EnableEgressRules` — Se desideri abilitare le regole di uscita per i gruppi di sicurezza, inserisci. `true` Se vuoi disabilitare le regole di uscita, inserisci. `false` 2. Attendi il completamento della distribuzione dello stack. 3. Negli output di creazione dello stack, prendi nota dell'endpoint del server SFTP. Questo valore ti servirà più tardi.	

Configura le risorse

Operazione	Description	Competenze richieste
Attiva la protezione da malware.	1. Accedi a Console di gestione AWS e apri la GuardDuty console. 2. Nel pannello di navigazione, scegli Malware Protection for S3.	Amministratore cloud, amministratore AWS

Operazione	Description	Competenze richieste
	3. Nella sezione Bucket protetti, scegli Abilita. 4. In Inserisci i dettagli del bucket S3, inserisci il nome del bucket Amazon S3. In alternativa, scegli Browse S3 per seleziona re un bucket. Scegli il `UploadBucket` bucket Amazon S3. 5. In Prefisso, seleziona Tutti gli oggetti nel bucket S3. 6. In Etichetta gli oggetti scansionati, seleziona Etichetta gli oggetti. 7. Nella sezione Accesso al servizio, scegli Crea e usa un nuovo ruolo di servizio. 8. Scegli Abilita .	
Aggiungi utenti al pool di utenti.	Aggiungi uno o più utenti al pool di utenti di Amazon Cognito. Per istruzioni, consulta Gestire gli utenti nel tuo pool di utenti nella documentazione di Amazon Cognito.	Amministratore cloud, amministratore AWS

Prova il server SFTP

Operazione	Description	Competenze richieste

Connect all'endpoint del server SFTP.	1. Immettete il seguente comando per ottenere l'endpoint del server SFTP, `<stack-name>` dov'è il nome del vostro stack: CloudFormation
	`TRANSFER_ENDPOINT=$(aws cloudformation describe-stacks \ --stack-name <stack-name> \ --query 'Stacks[0].Outputs[?OutputKey==`TransferServerEndpoint`].OutputValue' \ --output text)`
	2. Immettete il seguente comando per connettervi al server SFTP:
	`sftp user@company.com@$TRANSFER_ENDPOINT`
	3. Verifica di ricevere la seguente richiesta:
	`sftp>`
	Ciò indica che ora è possibile immettere
		Sviluppatore di app, amministratore cloud, architetto cloud, DevOps ingegnere

Operazione	Description	Competenze richieste
	comandi per eseguire trasferimenti di file.	

risoluzione dei problemi

Problema	Soluzione
L'autenticazione dell'utente non riesce	1. Verifica che l'utente esista nel pool di utenti di Amazon Cognito. 2. Verifica che l'utente sia confermato e abilitato. 3. Assicurati che l'utente fornisca la password corretta. 4. Verifica la presenza di errori nei CloudWatch registri. Per un elenco di AWS CLI comandi che possono aiutarti a eseguire queste procedure di risoluzione dei problemi, vedi Comandi utili per la risoluzione dei problemi nella sezione Informazioni aggiuntive.
L'autenticazione SFTP non riesce	1. Verifica che l'utente esista nel pool di utenti di Amazon Cognito. 2. Verifica che l'utente sia confermato e abilitato. 3. Assicurati che l'utente fornisca la password corretta. 4. Verifica che l'indirizzo IP sia consentito nel gruppo di sicurezza del AWS Transfer Family server.

Problema	Soluzione
	5. Verifica la presenza di errori nei log per la funzione di autenticazione Lambda. 6. Verifica che il ruolo IAM per AWS Transfer Family disponga delle autorizzazioni corrette. Per un elenco di AWS CLI comandi che possono aiutarti a eseguire questi passaggi di risoluzione dei problemi, consulta Comandi utili per la risoluzione dei problemi nella sezione Informazioni aggiuntive.
Accesso al caricamento dei file negato	1. Verifica che le autorizzazioni AWS KMS chiave siano configurate correttamente. 2. Consulta le politiche relative ai bucket di Amazon S3. 3. Verifica che le autorizzazioni dei ruoli IAM siano configurate correttamente. 4. Verifica che gli endpoint VPC siano configura ti correttamente 5. Verifica che il ruolo IAM per AWS Transfer Family disponga delle autorizzazioni corrette. Per un elenco di AWS CLI comandi che possono aiutarti a eseguire questi passaggi di risoluzione dei problemi, consulta Comandi utili per la risoluzione dei problemi nella sezione Informazioni aggiuntive.

Problema	Soluzione
Nessuna scansione antimalware	1. Abilita GuardDuty nel tuo account. 2. Abilita la protezione da malware per S3. 3. Verifica che la EventBridge regola Amazon sia attiva. 4. Verifica la presenza di errori nei log per la funzione Lambda di routing dei file. 5. Verifica che le notifiche siano configurate correttamente in Amazon SNS. Per un elenco di AWS CLI comandi che possono aiutarti a eseguire questi passaggi di risoluzione dei problemi, consulta Comandi utili per la risoluzione dei problemi nella sezione Informazioni aggiuntive.
Errori della funzione Lambda	1. Controlla la configurazione del VPC. 2. Verifica che gli endpoint VPC siano configura ti correttamente. 3. Verifica che il ruolo IAM per AWS Transfer Family disponga delle autorizzazioni corrette. 4. Verifica la presenza di errori nei CloudWatch log. 5. Verifica che le autorizzazioni AWS KMS chiave siano configurate correttamente. Per un elenco di AWS CLI comandi che possono aiutarti a eseguire queste procedure di risoluzione dei problemi, vedi Comandi utili per la risoluzione dei problemi nella sezione Informazioni aggiuntive.

Risorse correlate

- App web Transfer Family (documentazione Transfer Family)

Informazioni aggiuntive

Comandi utili per la risoluzione dei problemi

Controlla lo stato di uno CloudFormation stack:

```
aws cloudformation describe-stacks \
  --stack-name <STACK_NAME>
```

Elenca tutti gli utenti in un pool di utenti Amazon Cognito:

```
aws cognito-idp list-users \
  --user-pool-id <USER_POOL_ID>
```

Visualizza i log per le funzioni Lambda:

```
aws logs describe-log-groups \
  --log-group-name-prefix /aws/lambda/
```

Controlla lo stato di: GuardDuty

```
aws guardduty list-detectors
```

Controlla le regole del gruppo di sicurezza:

```
aws ec2 describe-security-groups \
  --group-ids <SECURITY_GROUP_ID> \
  --output table
```

Controlla lo stato del AWS Transfer Family server:

```
aws transfer describe-server \
  --server-id <SERVER_ID>
```

Elenca tutti i file in un bucket Amazon S3:

```
aws s3 ls s3://<BUCKET_NAME>/ \
  --recursive
```

Controlla lo stato di una EventBridge regola:

```
aws events describe-rule \
  --name <RULE_NAME>
```

Proteggi i dati sensibili nei CloudWatch registri utilizzando Amazon Macie

Anisha Salunkhe, Omar Franco e David Guardiola, Amazon Web Services

Riepilogo

Questo modello mostra come utilizzare Amazon Macie per rilevare automaticamente i dati sensibili in un gruppo di log di Amazon CloudWatch Logs implementando un flusso di lavoro completo di monitoraggio della sicurezza. La soluzione utilizza Amazon Data Firehose per lo streaming delle voci di CloudWatch log su Amazon Simple Storage Service (Amazon S3). Macie analizza periodicamente questo bucket alla ricerca di informazioni di identificazione personale (PII), dati finanziari e altri contenuti sensibili. L'infrastruttura viene distribuita tramite un AWS CloudFormation modello che fornisce tutte le configurazioni e le configurazioni necessarie. Servizi AWS

CloudWatch I log spesso contengono dati applicativi che possono includere inavvertitamente informazioni riservate sugli utenti. Ciò può creare rischi per la conformità e la sicurezza. Gli approcci tradizionali di monitoraggio dei registri non dispongono di funzionalità automatizzate di rilevamento dei dati sensibili. Ciò può rendere difficile identificare e rispondere alle potenziali esposizioni dei dati in tempo reale.

Questo modello aiuta i team di sicurezza e i responsabili della conformità a mantenere la riservatezza dei dati fornendo rilevazioni e avvisi automatici per i dati sensibili nei sistemi di registrazione. Questa soluzione consente una risposta proattiva agli incidenti tramite le notifiche di Amazon Simple Notification Service (Amazon SNS) e isola automaticamente i dati sensibili in un bucket Amazon S3 sicuro. Puoi personalizzare i modelli di rilevamento e integrare il flusso di lavoro con i processi operativi di sicurezza esistenti.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Autorizzazioni per creare uno stack CloudFormation
- Un gruppo di CloudWatch log Logs che desideri monitorare
- Un indirizzo e-mail attivo per ricevere notifiche da Amazon SNS
- Accesso a AWS CloudShell.

- (Facoltativo) Accesso a AWS Command Line Interface (AWS CLI), installato e configurato

Limitazioni

- Macie è soggetta a quote di servizio. Per ulteriori informazioni, consulta Quotas for Macie nella documentazione di Macie.

Architecture

Architettura Target

Il diagramma seguente mostra il flusso di lavoro per l'utilizzo di Macie per esaminare le voci di registro di CloudWatch Logs alla ricerca di dati sensibili.

Il flusso di lavoro mostra i seguenti passaggi:

1. Il gruppo CloudWatch Logs log genera i log, che sono soggetti al filtro di sottoscrizione.
2. Il filtro di sottoscrizione inoltra i log ad Amazon Data Firehose.
3. I log vengono crittografati con una chiave AWS Key Management Service (AWS KMS) quando passano attraverso il flusso di distribuzione di Amazon Data Firehose.
4. Il flusso di consegna consegna i log nel bucket di log esportati in Amazon S3.
5. Ogni giorno alle 4 del mattino, Amazon EventBridge avvia una AWS Lambda funzione che avvia una scansione Macie alla ricerca di dati sensibili nel bucket dei log esportati.
6. Se Macie identifica dati sensibili nel bucket, una funzione Lambda rimuove il registro dal bucket di log esportato e lo crittografa con un. AWS KMS key
7. La funzione Lambda isola i log che contengono dati sensibili nel bucket di isolamento dei dati.
8. L'identificazione di dati sensibili avvia un argomento di Amazon SNS.
9. Amazon SNS invia una notifica e-mail a un indirizzo e-mail configurato con informazioni sui log che contengono dati sensibili.

Risorse distribuite

Il CloudFormation modello distribuisce le seguenti risorse nel target Account AWS e: Regione AWS

- Due bucket Amazon S3:
  - Un bucket di log esportato per archiviare i dati dei log CloudWatch
  - Un bucket di isolamento dei dati per archiviare le informazioni sensibili
- Una EventBridge regola di Amazon che risponde alle scoperte di Macie
- AWS Lambda funzioni che avviano eventi ed esportano i log nei bucket Amazon S3
- Un argomento e un abbonamento ad Amazon SNS
- Uno stream Amazon Data Firehose
- Una sessione con Macie
- Un identificatore di dati personalizzato Macie
- Un filtro di sottoscrizione CloudWatch Logs
- AWS KMS chiavi per crittografare i log memorizzati nei bucket
- I ruoli e le politiche AWS Identity and Access Management (IAM) necessari per la soluzione

Tools (Strumenti)

Servizi AWS

- AWS CloudFormationti aiuta a configurare AWS le risorse, fornirle in modo rapido e coerente e gestirle durante tutto il loro ciclo di vita in Account AWS e. Regioni AWS
- Amazon CloudWatch Logs ti aiuta a centralizzare i log di tutti i tuoi sistemi e applicazioni, Servizi AWS così puoi monitorarli e archiviarli in modo sicuro.
- Amazon Data Firehose ti aiuta a distribuire dati di streaming in tempo reale ad altri Servizi AWS endpoint HTTP personalizzati e endpoint HTTP di proprietà di provider di servizi terzi supportati.
- Amazon EventBridge è un servizio di bus eventi senza server che ti aiuta a connettere le tue applicazioni con dati in tempo reale provenienti da una varietà di fonti. Ad esempio, fonti come AWS Lambda funzioni, endpoint di invocazione HTTP che utilizzano destinazioni API o bus di eventi in altre. Account AWS
- AWS Key Management Service (AWS KMS) consente di creare e controllare chiavi crittografiche per proteggere i dati.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.

- Amazon Macie ti aiuta a scoprire dati sensibili, offre visibilità sui rischi per la sicurezza dei dati e consente la protezione automatizzata contro tali rischi.
- Amazon Simple Notification Service (Amazon SNS) ti aiuta a coordinare e gestire lo scambio di messaggi tra editori e clienti, inclusi server Web e indirizzi e-mail.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.

Repository di codici

Il codice per questo pattern è disponibile nel repository GitHub sample-macie-for-securing-cloudwatch-logs.

Best practice

Segui le migliori pratiche riportate nella documentazione.CloudFormation CloudFormation

Epiche

Implementazione della soluzione

Operazione	Description	Competenze richieste
Clona il repository del codice.	Immettete il seguente comando per clonare il repository sulla vostra workstation locale: `git clone https://github.com/aws-samples/sample-macie-for-securing-cloudwatch-logs`	Sviluppatore di app
(Facoltativo) Modifica il modello. CloudFormation	1. Apri il file main.yaml. 2. Personalizzate il modello effettuando una delle seguenti operazioni:	Sviluppatore di app

Operazione	Description	Competenze richieste
	• È possibile rinominare le risorse. • Nella `parameter` sezione è possibile modificare i valori predefiniti. • È possibile modificare il modello di filtro degli abbonamenti. Per ulteriori informazioni, consulta Filtri di sottoscrizione a livello di gruppo di log nella documentazione di CloudWatch Logs. 3. Salva e chiudi il file main.yaml.	

Operazione	Description	Competenze richieste

Opzione 1: distribuzione tramite script con parametri della riga di comando.	Immetti il seguente comando per distribuire la soluzione utilizzando i parametri della riga di comando, dove il valore `enable-macie` è `true` solo se Amazon Macie non è già abilitato:
	`./scripts/test-macie-solution.sh --deploy-stack \`
	`  --stack-name <stack name> \`
	`  --email <email address> \`
	`  --enable-macie <true or false> \`
	`  --region <region> \`
	`  --resource-name <prefix for all resources> \`
	`  --bucket-name <bucket name>`
		Informazioni generali su AWS

Operazione	Description	Competenze richieste

Opzione 2: implementazione tramite script con variabili di ambiente.	1. Inserisci i seguenti comandi per definire le variabili di ambiente, il cui valore `ENABLE_MACIE` è `true` solo se Amazon Macie non è già abilitato:
	`export STACK_NAME=<stack name>`
	`export SNS_EMAIL=<email address>`
	`export ENABLE_MACIE=<true or false>`
	`export REGION=<region>`
	`export RESOURCE_NAME=<prefix for all resources>`
	`export BUCKET_NAME=<bucket name>`
	2. Inserisci il seguente comando per convalidare i parametri prima della distribuzione:
	`./scripts/test-macie-solution.sh \ --validate-params \ --email <email address> \ --region <region>`
	3. Immettere il comando seguente per distribuire la soluzione:
		Informazioni generali su AWS

Operazione	Description	Competenze richieste
	`./scripts/test-macie-solution.sh --deploy-stack`	

Operazione	Description	Competenze richieste
Opzione 3: distribuisci utilizzando. AWS CLI	Immetti il seguente comando per distribuire la soluzione utilizzando AWS CLI, dove il valore per `EnableMacie` è `true` solo se Amazon Macie non è già abilitato: `aws cloudformation create-stack \ --region us-east-1 \ --stack-name macie-for-securing-cloudwatch-logs \ --template-body file://app/main.yml \ --capabilities CAPABILITY_IAM \ --parameters \ ParameterKey=ResourceName,ParameterValue=<prefix for all resources> \ ParameterKey=BucketName,ParameterValue=<bucket name> \ ParameterKey=LogGroupName,ParameterValue=<path for log group> \ ParameterKey=SNSTopicEndpointEmail,ParameterValue=<email address> \ ParameterKey=EnableMacie,ParameterValue=<true or false>`	

Operazione	Description	Competenze richieste
Opzione 4: distribuzione tramite. Console di gestione AWS	1. Apri la AWS CloudFormation console. 2. Nella barra di navigazione nella parte superiore dello schermo, scegli in cui Regione AWS creare lo stack. 3. Dalla pagina Stack, scegli Crea stack in alto a destra, poi scegli Con nuove risorse (standard). 4. Nella pagina Crea stack, per Prerequisito - Prepara modello, scegli Scegli un modello esistente. 5. In Specificare modello, scegli Carica un file modello, quindi carica il modello main.yaml dal tuo repository clonato. 6. Scegli Next (Successivo). 7. Nella pagina Specificare i dettagli dello stack, nella casella Nome dello stack, inserisci un nome per lo stack. 8. Nella sezione Parametri, specificate i valori per i seguenti parametri del modello. • `ResourceName` : prefisso per tutte le risorse	Informazioni generali su AWS

Operazione	Description	Competenze richieste
	• `BucketName` : nome univoco per il bucket Amazon S3 • `LogGroupName` : nome del gruppo di log per Logs CloudWatch • `SNSTopicEndpointEmail` : indirizzo e-mail per le notifiche • `EnableMacie` : Imposta `true` se Macie non è già abilitato • (Facoltativo)`Region`: il Regione AWS punto in cui desideri distribuire lo stack. • (Facoltativo)`TemplatePath` : percorso verso il modello CloudFormation 9. Scegli Next (Successivo). 10. Per Funzionalità, scegli Riconosco che questo modello può creare risorse IAM per specificare che desideri utilizzare le risorse IAM nel modello. 11. Scegli Next (Successivo). 12. Nella pagina Rivedi e crea, rivedi i dettagli dello stack. 13. Scegli Invia per avviare lo stack.	

Operazione	Description	Competenze richieste
Monitora lo stato della distribuzione e conferma la distribuzione.	1. Immettete il seguente comando per monitorare lo stato della distribuzione: `./scripts/test-mac ie-solution.sh \ --deployment-statu s \ --stack-name <stack name>` **Note** Puoi anche monitorare l'avanzamento e lo stato della creazione dello stack nella scheda Eventi per il tuo nuovo stack. Per ulteriori informazioni, consulta Monitorare l'avanzamento dello stack. 2. Quando lo stato cambia in CREATE_COMPLETE, esamina gli output dello stack per le informazioni sulle risorse.	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Conferma l'abbonamento ad Amazon SNS.	Segui le istruzioni in Conferma l'abbonamento ad Amazon SNS nella documentazione di Amazon SNS per confermar e l'abbonamento ad Amazon SNS.	Sviluppatore di app

Test della soluzione

Operazione	Description	Competenze richieste

Opzione 1: test con report automatizzati.	Se hai utilizzato il nome dello stack predefinito, inserisci il seguente comando per testare la soluzione:
	`./scripts/test-macie-solution.sh \ --full-test`
	Se hai usato un nome di stack personalizzato, inserisci il seguente comando per testare la soluzione:
	`./scripts/test-macie-solution.sh \ --full-test \ --stack-name <stack name>`
	Se hai usato un nome di stack personalizzato e parametri personalizzati, inserisci il
		Informazioni generali su AWS

Operazione	Description	Competenze richieste

	seguente comando per testare la soluzione:
	
	`./scripts/test-macie-solution.sh --full-test \`
	`  --stack-name <stack name> \`
	`  --region <region> \`
	`  --log-group <log group path>`	

Operazione	Description	Competenze richieste
Opzione 2: test con convalida mirata.	1. Immettete il seguente comando per generare dati di test con informazioni sensibili: `./scripts/test-macie-solution.sh \ --generate-test-data \ --stack-name <stack name>` Questo comando funziona nel modo seguente: • Crea CloudWatch voci di registro che contengono modelli di dati sensibili realistici, tra cui dipendenti, brevetti IDs IDs, numeri di carte di credito, numeri di previdenza sociale e indirizzi e-mail • Genera voci di registro sensibili e non sensibili per test completi • Fornisce una registrazione dettagliata del processo di generazione dei dati di test 2. Immettete il seguente comando per verificare il flusso della pipeline di dati:	Informazioni generali su AWS

Operazione	Description	Competenze richieste

	``
	./scripts/test-mac
	ie-solution.sh \
	  --verify-pipeline \
	  --stack-name <stack
	 name>
	``
	Questo comando funziona nel modo seguente:
	• Conferma che le voci dei CloudWatch log vengono trasmesse in streaming ad Amazon Data Firehose
	• Verifica che i dati di log siano stati consegnati al bucket Amazon S3 con una crittografia adeguata
	• Verifica che lo storage di oggetti Amazon S3 abbia la struttura dei prefissi corretta
	• Verifica lo stato di crittografia degli oggetti archiviati
	• Monitora la tempistica del flusso di dati e fornisce periodi di attesa per l'elaborazione
	3. Immettete il seguente comando per avviare il processo di classificazione di Macie:	

Operazione	Description	Competenze richieste

	`./scripts/test-macie-solution.sh \ --trigger-macie-job \ --stack-name <stack name>`
	
	Questo comando funziona nel modo seguente:
	• Attiva manualmente il processo di classificazione Macie tramite una funzione Lambda
	• Monitora lo stato di esecuzione del lavoro e fornisce feedback
	• Convalida la disponibilità del servizio Macie prima dell'esecuzione
	• Gestisce i casi in cui Macie non è abilitato nell'account
	• Fornisce risultati dettagliati sull'esecuzione del lavoro
	4. Immettete il seguente comando per convalidare gli avvisi e l'isolamento dei dati:
	
	`./scripts/test-macie-solution.sh \ --verify-alerts \`	

Operazione	Description	Competenze richieste
	`--stack-name <stack name>` Questo comando funziona nel modo seguente: • Conferma che EventBridge le regole sono configurate e attive correttamente • Convalida la configurazione dell'argomento Amazon SNS e lo stato dell'abbonamento • Verifica la configurazione e le autorizzazioni del bucket di isolamento dei dati • Monitora lo spostamento dei dati sensibili verso il bucket di isolamento • Verifica la funzionalità completa del flusso di lavoro di avviso	

Eliminazione

Operazione	Description	Competenze richieste
Opzione 1: eseguire la pulizia automatica.	Se hai utilizzato il nome dello stack predefinito, inserisci il seguente comando per eliminare lo stack:	Informazioni generali su AWS

Operazione	Description	Competenze richieste

	`./scripts/cleanup-macie-solution.sh \ --full-cleanup`
	
	Se hai usato un nome di stack personalizzato, inserisci il seguente comando per eliminare lo stack:
	
	`./scripts/cleanup-macie-solution.sh \ --full-cleanup \ --stack-name <stack name>`
	
	Se hai usato un nome di stack personalizzato e parametri personalizzati, inserisci il seguente comando per eliminare lo stack:
	
	`./scripts/cleanup-macie-solution.sh \ --full-cleanup \ --stack-name <stack name> \ --region <region> \ --disable-macie <true or false>`	

Operazione	Description	Competenze richieste

Opzione 2: eseguire step-by-step la pulizia.	1. Immettere il seguente comando per interrompere i processi attivi:
	`./scripts/cleanup-macie-solution.sh \ --stop-processes \ --stack-name <stack name>`
	Questo comando funziona nel modo seguente:
	• Disattiva EventBridge le regole per impedire l'esecuzione di nuovi lavori
	• Interrompe tutti i processi di classificazione di Macie attualmente in esecuzione
	• Annulla le esecuzioni dei job di Macie in sospeso
	• Cancella tutti i messaggi Amazon SNS in sospeso nella coda
	• Fornisce aggiornamenti sullo stato per ogni processo interrotto
	2. Inserisci il seguente comando per svuotare i bucket Amazon S3:
	`./scripts/cleanup-macie-solution.sh \`
		Informazioni generali su AWS

Operazione	Description	Competenze richieste

	`--empty-buckets \`
	`--stack-name <stack name>`
	
	Questo comando funziona nel modo seguente:
	• Rimuove tutti gli oggetti dal bucket per i log CloudWatch
	• Rimuove tutti gli oggetti dal bucket di isolamento dei dati Amazon S3
	• Elimina tutti i caricamenti multiparte incompleti
	• Gestisce gli oggetti con versione se il controllo delle versioni di Amazon S3 è abilitato
	• Fornisce aggiornamenti sul conteggio degli oggetti e sull'avanzamento dell'eliminazione
	3. Immettete il seguente comando per eliminare lo CloudFormation stack:
	
	`./scripts/cleanup-macie-solution.sh \`
	`--delete-stack \`
	`--stack-name <stack name>`
	
	Questo comando funziona nel modo seguente:	

Operazione	Description	Competenze richieste
	• Avvia il processo di eliminazione dello CloudFormation stack • Monitora l'avanzamento dell'eliminazione con aggiornamenti di stato in tempo reale • Gestisce gli errori di eliminazione con una segnalazione dettagliata degli errori • Attende la rimozione completa dello stack prima di procedere • Fornisce la cronologia degli eventi dello stack per la risoluzione dei problemi 4. Immettete il seguente comando per ripulire le risorse di Macie: `./scripts/cleanup-macie-solution.sh \ --cleanup-macie \ --stack-name <stack name>` Questo stack esegue le seguenti operazioni: • Rimuove gli identificatori di dati personalizzati creati dalla soluzione	

Operazione	Description	Competenze richieste
	• Pulisce tutti gli artefatti e i risultati del lavoro di Macie rimanenti • Disattiva la sessione Macie se è stata abilitata dallo stack • Gestisce i casi in cui le risorse Macie vengono condivise con altre applicazioni • Fornisce lo stato di pulizia dettagliato per ogni componente Macie	

Operazione	Description	Competenze richieste

Verifica la pulizia.	1. Immettete il seguente comando per verificare che lo stack sia stato eliminato:
	`aws cloudformation describe-stacks \ --stack-name <stack name> \ --region <region>`
	2. Inserisci il seguente comando per verificare che i bucket Amazon S3 siano stati eliminati:
	`aws s3 ls | grep macie`
	3. Inserisci il seguente comando per verificare che gli identificatori di dati personalizzati Macie siano stati rimossi:
	`aws macie2 list-custom-data-identifiers \ --region <region>`
	4. Inserisci il seguente comando per verificare la presenza di eventuali risorse rimanenti:
	`./scripts/cleanup-macie-solution.sh \ --verify-cleanup \`
		Informazioni generali su AWS

Operazione	Description	Competenze richieste
	`--stack-name <stack name>`	

risoluzione dei problemi

Problema	Soluzione
CloudFormation lo stato dello stack mostra CREATE_FAILED.	Il CloudFormation modello è configurato per pubblicare i log in Logs. CloudWatch Puoi visualizzare i log in in Console di gestione AWS modo da non doverti connettere alla tua EC2 istanza Amazon. Per ulteriori informazioni, consulta Visualizzare CloudFormation i log nella console (post delAWS blog).
CloudFormation `delete-stack` il comando fallisce.	Alcune risorse devono essere vuote prima di poter essere eliminate. Ad esempio, devi eliminare tutti gli oggetti in un bucket Amazon S3 o rimuovere tutte le istanze in un gruppo di EC2 sicurezza Amazon prima di poter eliminare il bucket o il gruppo di sicurezza. Per ulteriori informazioni, consulta Delete stack fail nella documentazione di Amazon S3.
Errore durante l'analisi di un parametro.	Quando usi la console AWS CLI o la CloudFormation console per passare un valore, aggiungi le virgolette.

Risorse correlate

- Procedure ottimali di architettura per lo storage (AWS sito Web)
- Sintassi del pattern di filtro per filtri metrici, filtri di sottoscrizione, eventi di log dei filtri e Live Tail (documentazione di CloudWatch Logs)

- Progettazione e implementazione della registrazione e del monitoraggio con Amazon CloudWatch (AWS Prescriptive Guidance)
- Risoluzione dei problemi CloudFormation (documentazione)CloudFormation

Invia avvisi da AWS Network Firewall a un canale Slack

Venki Srivatsav e Aromal Raj Jayarajan, Amazon Web Services

Riepilogo

Questo modello descrive come implementare un firewall utilizzando il Network Firewall di Amazon Web Services (AWS) con il modello di distribuzione distribuito e come propagare gli avvisi generati da AWS Network Firewall su un canale Slack configurabile.

Gli standard di conformità come Payment Card Industry Data Security Standard (PCI DSS) richiedono l'installazione e la manutenzione di un firewall per proteggere i dati dei clienti. Nel cloud AWS, un cloud privato virtuale (VPC) è considerato uguale a una rete fisica nel contesto di questi requisiti di conformità. È possibile utilizzare Network Firewall per monitorare il traffico di rete tra VPCs e proteggere i carichi di lavoro eseguiti in ambienti VPCs regolati da uno standard di conformità. Network Firewall blocca l'accesso o genera avvisi quando rileva l'accesso non autorizzato da parte di altri utenti dello VPCs stesso account. Tuttavia, Network Firewall supporta un numero limitato di destinazioni per l'invio degli avvisi. Queste destinazioni includono bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3), gruppi di log CloudWatch Amazon e flussi di distribuzione Amazon Data Firehose. Qualsiasi ulteriore azione su queste notifiche richiede un'analisi offline utilizzando Amazon Athena o Amazon Kinesis.

Questo modello fornisce un metodo per propagare gli avvisi generati da Network Firewall su un canale Slack configurabile per ulteriori azioni quasi in tempo reale. Puoi anche estendere la funzionalità ad altri meccanismi di avviso come PagerDuty Jira ed e-mail. (Queste personalizzazioni non rientrano nell'ambito di questo modello).

Prerequisiti e limitazioni

Prerequisiti

- Canale Slack (vedi Guida introduttiva nel centro assistenza Slack)
- Privilegi necessari per inviare un messaggio al canale
- L'URL dell'endpoint Slack con un token API (seleziona l'app e scegli un webhook in entrata per visualizzarne l'URL; per maggiori informazioni, consulta Creazione di un webhook in entrata nella documentazione dell'API Slack)
- Un'istanza di test Amazon Elastic Compute Cloud (Amazon EC2) nelle sottoreti dei carichi di lavoro
- Regole di test in Network Firewall

- Traffico reale o simulato per attivare le regole del test
- Un bucket S3 per contenere i file sorgente da distribuire

Limitazioni

- Attualmente questa soluzione supporta solo un singolo intervallo CIDR (Classless Inter-Domain Routing) come filtro per origine e destinazione. IPs

Architecture

Stack tecnologico Target

- Un VPC
- Quattro sottoreti (due per il firewall e due per i carichi di lavoro)
- Internet Gateway
- Quattro tabelle di routing con regole
- Bucket S3 utilizzato come destinazione di avviso, configurato con una policy del bucket e impostazioni degli eventi per eseguire una funzione Lambda
- Funzione Lambda con ruolo di esecuzione, per inviare notifiche Slack
- Segreto di AWS Secrets Manager per l'archiviazione dell'URL Slack
- Firewall di rete con configurazione degli avvisi
- Canale Slack

Tutti i componenti tranne il canale Slack sono forniti dai CloudFormation modelli e dalla funzione Lambda forniti con questo modello (vedi la sezione Codice).

Architettura Target

Questo modello configura un firewall di rete decentralizzato con integrazione Slack. Questa architettura è costituita da un VPC con due zone di disponibilità. Il VPC include due sottoreti protette e due sottoreti firewall con endpoint firewall di rete. Tutto il traffico in entrata e in uscita dalle sottoreti protette può essere monitorato creando politiche e regole del firewall. Il firewall di rete è configurato per inserire tutti gli avvisi in un bucket S3. Questo bucket S3 è configurato per chiamare una funzione Lambda quando riceve un evento. `put` La funzione Lambda recupera l'URL Slack configurato da Secrets Manager e invia il messaggio di notifica all'area di lavoro Slack.

Per ulteriori informazioni su questa architettura, consulta il post sul blog AWS Deployment models for AWS Network Firewall.

Tools (Strumenti)

Servizi AWS

- AWS Network Firewall è un firewall di rete a stato gestito e un servizio di rilevamento e prevenzione delle intrusioni VPCs nel cloud AWS. Puoi utilizzare Network Firewall per filtrare il traffico lungo il perimetro del tuo VPC e proteggere i tuoi carichi di lavoro su AWS.
- AWS Secrets Manager è un servizio per l'archiviazione e il recupero delle credenziali. Utilizzando Secrets Manager, puoi sostituire le credenziali codificate nel codice, comprese le password, con una chiamata API a Secrets Manager per recuperare il segreto a livello di codice. Questo pattern utilizza Secrets Manager per memorizzare l'URL di Slack.
- Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è un servizio di storage di oggetti. È possibile utilizzare Amazon S3 per memorizzare e recuperare qualsiasi volume di dati, in qualunque momento e da qualunque luogo tramite il Web. Questo modello utilizza Amazon S3 per archiviare i CloudFormation modelli e lo script Python per la funzione Lambda. Utilizza anche un bucket S3 come destinazione degli avvisi del firewall di rete.
- AWS ti CloudFormation aiuta a modellare e configurare le tue risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle per tutto il loro ciclo di vita. Puoi utilizzare un modello per descrivere le tue risorse e le loro dipendenze e lanciarle e configurarle insieme come uno stack, invece di gestire le risorse singolarmente. Questo modello utilizza AWS CloudFormation per implementare automaticamente un'architettura distribuita per Firewall Manager.

Codice

Il codice per questo pattern è disponibile su GitHub, nel repository Network Firewall Slack Integration. Nella `src` cartella del repository troverai:

- Un set di CloudFormation file in formato YAML. Questi modelli vengono utilizzati per fornire i componenti per questo modello.
- Un file sorgente Python (`slack-lambda.py`) per creare la funzione Lambda.
- Un pacchetto di distribuzione dell'archivio.zip (`slack-lambda.py.zip`) per caricare il codice della funzione Lambda.

Per utilizzare questi file, segui le istruzioni nella sezione successiva.

Epiche

Configura il bucket S3

Operazione	Description	Competenze richieste
Crea un bucket S3.	1. Accedi alla Console di gestione AWS e apri la console Amazon S3 all'indirizzo. https://console.aws.amazon.com/s3/ 2. Scegli o crea un bucket S3 per ospitare il codice. Il nome di un bucket S3 è unico a livello globale e lo spazio dei nomi è condiviso da tutti gli account AWS. Il nome del bucket S3 non può includere barre iniziali. Ti consigliamo di utilizzare un prefisso per organizzare il codice per questo pattern. Per ulteriori informazioni, consulta Creazione di un bucket nella documentazione di Amazon S3.	Sviluppatore di app, proprietario dell'app, amministratore del cloud
Carica i CloudFormation modelli e il codice Lambda.	1. Scarica i seguenti file dal GitHub repository per questo pattern: • `base.yml` • `igw-ingress-route.yml`	Sviluppatore di app, proprietario dell'app, amministratore del cloud

Operazione	Description	Competenze richieste
	• `slack-lambda.py` • `slackLambda.yml` • `decentralized-deployment.yml` • `protected-subnet-route.yml` • `slack-lambda.py.zip` 2. Carica i file nel bucket S3 che hai creato.	

Implementa il modello CloudFormation

Operazione	Description	Competenze richieste
Avvia il CloudFormation modello.	Apri la CloudFormation console AWS nella stessa regione AWS del bucket S3 e distribuisci il modello. `base.yml` Questo modello crea le risorse AWS e le funzioni Lambda richieste per la trasmissione degli avvisi al canale Slack. Per ulteriori informazioni sulla distribuzione dei CloudFormation modelli, consulta Creazione di uno stack sulla CloudFormation console AWS nella CloudFormation documentazione.	Sviluppatore di app, proprietario dell'app, amministratore del cloud

Operazione	Description	Competenze richieste
Completa i parametri nel modello.	Specificate il nome dello stack e configurate i valori dei parametri. Per un elenco dei parametri, delle relative descrizioni e dei valori predefiniti, consultate CloudFormation i parametri nella sezione Informazioni aggiuntive.	Sviluppatore di app, proprietario dell'app, amministratore del cloud
Creare lo stack.	1. Rivedi i dettagli dello stack e aggiorna i valori in base ai requisiti dell'ambiente. 2. Scegli Crea stack per distribuire il modello.	Sviluppatore di app, proprietario dell'app, amministratore del cloud

Verifica la soluzione

Operazione	Description	Competenze richieste
Testa la distribuzione.	Utilizza la CloudFormation console AWS o l'AWS Command Line Interface (AWS CLI) per verificare che le risorse elencate nella sezione dello stack tecnologico di Target siano state create. Se il CloudFormation modello non riesce a essere distribuito correttamente, controlla i valori che hai fornito per i `pAvailabilityZone1`	Sviluppatore di app, proprietario dell'app, amministratore del cloud

Operazione	Description	Competenze richieste
	parametri and. `pAvailabilityZone2` Questi dovrebbero essere appropriati per la regione AWS in cui stai distribuendo la soluzione. Per un elenco delle zone di disponibilità per ogni regione, consulta Regioni e zone nella EC2 documentazione di Amazon.	

Operazione	Description	Competenze richieste

Funzionalità di test.	1. Apri la EC2 console Amazon all'indirizzo https://console.aws.amazon.com/ec2/.
	
	2. Crea un' EC2 istanza in una delle sottoreti protette. Scegli un AMI Amazon Linux 2 (HVM) da usare come server HTTPS. Per istruzioni, consulta Launch an istance nella EC2 documentazione di Amazon.
	
	Note
	Amazon Linux 2 sta per terminare il supporto. Per ulteriori informazioni, consulta Amazon Linux 2 FAQs.
	
	3. Utilizza i seguenti dati utente per installare un server Web sull' EC2 istanza:
	
	`#!/bin/bash`
	`yum install httpd -y`
	`systemctl start httpd`
	`systemctl stop firewalld`
	`cd /var/www/html`
		Sviluppatore di app, proprietario dell'app, amministratore del cloud

Operazione	Description	Competenze richieste

	`echo "Hello!! this is a NFW alert test page, 200 OK" > index.html`
	
	4. Crea le seguenti regole del firewall di rete:
	
	Regola apolide:
	
	`Source: 0.0.0.0/0`
	`Destination 10.0.3.65 /32 (private IP of the EC2 instance)`
	`Action: Forward`
	
	Regola statica:
	
	`Protocol: HTTP`
	`Source ip/port: Any / Any`
	`Destination ip/port: Any /Any`
	
	5. Ottieni l'IP pubblico del server web che hai creato nel passaggio 3.
	
	6. Accedi all'IP pubblico in un browser. Dovresti vedere il seguente messaggio nel browser:
	
	`Hello!! this is a NFW alert test page, 200 OK`
	
	Riceverai anche una notifica nel canale Slack. La notifica	

Operazione	Description	Competenze richieste
	potrebbe subire ritardi, a seconda della dimensione del messaggio. A scopo di test, valuta la possibilità di fornire un filtro CIDR non troppo stretto (ad esempio, un valore CIDR con /32 sarebbe considerato troppo stretto e /8 sarebbe troppo ampio). Per ulteriori informazioni, consulta la sezione Comportamento del filtro in Informazioni aggiuntive.	

Risorse correlate

- Modelli di distribuzione per AWS Network Firewall (post sul blog AWS)
- Policy di AWS Network Firewall (documentazione AWS)
- Integrazione Slack con Network Firewall (GitHub repository)
- Crea uno spazio di lavoro Slack (centro assistenza Slack)

Informazioni aggiuntive

CloudFormation parameters

Parametro	Description	Valore predefinito o di esempio
`pVpcName`	Il nome del VPC da creare.	Ispezione
`pVpcCidr`	L'intervallo CIDR per il VPC da creare.	10.0.0.0/16

pVpcInstanceTenancy	Come vengono distribuite EC2 le istanze sull'hardware fisico. Le opzioni sono `default` (locazione condivisa) o `dedicated` (locazione singola).	default
pAvailabilityZone1	La prima zona di disponibilità per l'infrastruttura.	us-east-2a
pAvailabilityZone2	La seconda zona di disponibilità per l'infrastruttura.	us-east-2b
pNetworkFirewallSubnet1Cidr	L'intervallo CIDR per la prima sottorete del firewall (minimo /28).	10.0.1.0/24
pNetworkFirewallSubnet2Cidr	L'intervallo CIDR per la seconda sottorete del firewall (minimo /28).	10.0.2.0/24
pProtectedSubnet1Cidr	L'intervallo CIDR per la prima sottorete protetta (carico di lavoro).	10.0.3.0/24
pProtectedSubnet2Cidr	L'intervallo CIDR per la seconda sottorete protetta (carico di lavoro).	10.0.4.0/24
pS3BucketName	Il nome del bucket S3 esistente in cui hai caricato il codice sorgente Lambda.	us-w2- yourname-lambda-functions
pS3KeyPrefix	Il prefisso del bucket S3 in cui hai caricato il codice sorgente Lambda.	aod-test
pAWSSecretName4Slack	Il nome del segreto che contiene l'URL di Slack.	SlackEnpoint-Cfn

`pSlackChannelName`	Il nome del canale Slack che hai creato.	alcune notifiche di nome
`pSlackUserName`	nome utente Slack.	Utente Slack
`pSecretKey`	Questa può essere una chiave qualsiasi. Ti consigliamo di utilizzare l'impostazione predefinita.	WebHookURL
`pWebHookUrl`	Il valore dell'URL Slack.	https://hooks.slack.com/services/T??? 9 T?? /A031885 JRM7 /9D4Y?????
`pAlertS3Bucket`	Il nome del bucket S3 da utilizzare come destinazione degli avvisi del firewall di rete. Questo bucket verrà creato per te.	us-w2- yourname-security-aod-alerts
`pSecretTagName`	Il nome del tag per il segreto.	AppName
`pSecretTagValue`	Il valore del tag per il nome del tag specificato.	LambdaSlackIntegration
`pdestCidr`	Il filtro per l'intervallo CIDR di destinazione. Per ulteriori informazioni, consultat e la sezione successiva, Comportamento del filtro.	10.0.0.0/16
`pdestCondition`	Un contrassegno per indicare se escludere o includere la corrispondenza di destinazi one. Per ulteriori informazioni, consulta la sezione successiva I valori validi sono `include` e `exclude`.	includere

`psrcCidr`	Il filtro per l'intervallo CIDR di origine da avvisare. Per ulteriori informazioni, consulta la sezione successiva	118.2.0.0/16
`psrcCondition`	Il contrassegno per escludere o includere la corrispon denza di origine. Per ulteriori informazioni, consulta la sezione successiva	includere

Comportamento del filtro

Se non hai configurato alcun filtro in AWS Lambda, tutti gli avvisi generati vengono inviati al tuo canale Slack. L'origine e la destinazione IPs degli avvisi generati vengono confrontati con gli intervalli CIDR configurati durante la distribuzione del modello. CloudFormation Se viene trovata una corrispondenza, viene applicata la condizione. Se l'origine o la destinazione rientrano nell'intervallo CIDR configurato e almeno una di esse è configurata con la condizione`include`, viene generato un avviso. Le tabelle seguenti forniscono esempi di valori, condizioni e risultati CIDR.

	CIDR configurato	Avviso IP	Configurato	Alert
Origine	10.0.0.0/16	10.0.0.25	includere	Sì
Destinazione	100,0,0,0/16	20220,0,13	includere	

	CIDR configurato	Avviso IP	Configurato	Alert
Origine	10.0.0.0/16	10.0.0.25	exclude	No
Destinazione	100,0,0,0/16	20220,0,13	includere	

	CIDR configurato	Avviso IP	Configurato	Alert
Origine	10.0.0.0/16	10.0.0.25	includere	Sì

Destinazione	100,0,0,0/16	100,0,0,13	includere	

	CIDR configurato	Avviso IP	Configurato	Alert
Origine	10.0.0.0/16	90.0.0.25	includere	Sì
Destinazione	Null	20220,0,13	includere	

	CIDR configurato	Avviso IP	Configurato	Alert
Origine	10.0.0.0/16	90.0.0.25	includere	No
Destinazione	100,0,0,0/16	20220,0,13	includere	

Invia attributi personalizzati ad Amazon Cognito e inseriscili in token

Carlos Alessandro Ribeiro e Mauricio Mendoza, Amazon Web Services

Riepilogo

L'invio di attributi personalizzati a un processo di autenticazione Amazon Cognito può fornire un contesto aggiuntivo a un'applicazione, consentire controlli di accesso più granulari e semplificare la gestione dei profili utente e dei requisiti di autenticazione. Queste funzionalità sono utili in un'ampia gamma di applicazioni e scenari e possono aiutarti a migliorare la sicurezza e la funzionalità complessive di un'applicazione.

Questo modello mostra come inviare attributi personalizzati a un processo di autenticazione Amazon Cognito quando un'applicazione deve fornire un contesto aggiuntivo al token di accesso o al token di identità (ID). Utilizzi Node.js come applicazione di backend. L'applicazione autentica un utente da un pool di utenti di Amazon Cognito e trasmette gli attributi personalizzati necessari per la generazione di token. Puoi utilizzare i AWS Lambda trigger per Amazon Cognito per personalizzare il processo di autenticazione senza grandi personalizzazioni del codice o sforzi significativi.

> **Important**
>
> Il codice e gli esempi di questo modello non sono consigliati per i carichi di lavoro di produzione perché sono destinati esclusivamente a scopi dimostrativi. Per i carichi di lavoro di produzione, è richiesta una configurazione aggiuntiva sul lato client. Utilizzate questo modello come riferimento solo per proof-of-concept scopi pilota o a scopo pilota.

Prerequisiti e limitazioni

Prerequisiti

- Un attivo Account AWS
- Autorizzazioni per creare e gestire pool di utenti e funzioni di Amazon Cognito AWS Lambda
- AWS Command Line Interface (AWS CLI), installato e configurato
- Un ambiente di sviluppo integrato (IDE) che supporta Node.js

- Node.js versione 18 o successiva, installata
- npm versione 8 o successiva, installata
- TypeScript, installato

Limitazioni

- Questo modello non è applicabile all'integrazione delle applicazioni tramite il flusso di autenticazione Client Credentials.
- Il trigger di generazione del pre-token può aggiungere o modificare solo alcuni attributi del token di accesso e del token di identità. Per ulteriori informazioni, consulta Trigger Lambda prima della generazione di token nella documentazione di Amazon Cognito.

Architecture

Architettura Target

Il diagramma seguente mostra l'architettura di destinazione per questo modello. Mostra anche come l'applicazione Node.js potrebbe funzionare con un backend per aggiornare i database. Tuttavia, gli aggiornamenti del database di backend non rientrano nell'ambito di questo modello.

Il diagramma mostra il flusso di lavoro seguente:

1. L'applicazione Node.js emette un token di accesso con attributi personalizzati al pool di utenti di Amazon Cognito.
2. Il pool di utenti di Amazon Cognito avvia la funzione Lambda per la generazione di pre-token, che personalizza i token di accesso e ID.
3. L'applicazione Node.js effettua una chiamata API tramite Amazon API Gateway.

> **Note**
>
> Gli altri componenti architettonici mostrati in questa architettura sono solo a titolo di esempio e non rientrano nell'ambito di questo modello.

Automazione e scalabilità

Puoi automatizzare il provisioning di pool di utenti AWS Lambda , funzioni, istanze di database e altre risorse di Amazon Cognito AWS CloudFormationutilizzando Terraform o qualsiasi strumento Infrastructure as Code (IaC) supportato. AWS Cloud Development Kit (AWS CDK)HashiCorp Se desideri scalare le tue implementazioni, utilizza pipeline di integrazione continua e distribuzione continua (CI/CD), che aiutano a prevenire gli errori associati alle distribuzioni manuali.

Tools (Strumenti)

Servizi AWS

- Amazon API Gateway ti aiuta a creare, pubblicare, gestire, monitorare e proteggere REST, HTTP e WebSocket APIs su qualsiasi scala.
- Amazon Cognito fornisce autenticazione, autorizzazione e gestione degli utenti per app Web e mobili.
- Amazon Elastic Container Service (Amazon ECS) è un servizio rapido e scalabile di gestione dei container che ti aiuta a eseguire, arrestare e gestire container in un cluster.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.
- AWS SDK per JavaScriptfornisce un' JavaScript API per. Servizi AWSÈ possibile utilizzarlo per creare librerie o applicazioni per Node.js o per il browser.

Altri strumenti

- Node.js è un ambiente di JavaScript runtime basato sugli eventi progettato per la creazione di applicazioni di rete scalabili.
- npm è un registro software che viene eseguito in un ambiente Node.js e viene utilizzato per condividere o prendere in prestito pacchetti e gestire la distribuzione di pacchetti privati.

Best practice

Ti consigliamo di implementare le seguenti best practice:

- Segreti e dati sensibili: non archiviate segreti o dati sensibili all'interno dell'applicazione. Utilizzate un sistema esterno da cui l'applicazione può estrarre i dati AWS AppConfigGestione dei segreti AWS, ad esempio AWS Systems Manager Parameter Store.

- Distribuzione standardizzata: utilizza CI/CD le pipeline per distribuire le applicazioni. È possibile utilizzare servizi come e. AWS CodeBuildAWS CodePipeline
- Scadenza del token: imposta una data di scadenza breve per il token di accesso.
- Utilizza una connessione sicura: tutte le comunicazioni tra l'applicazione client e il backend devono essere crittografate utilizzando SSL/TLS. Usa AWS Certificate Manager (ACM) per generare e gestire SSL/TLS certificati e usa Amazon CloudFront o Elastic Load Balancing per gestire SSL/TLS la terminazione.
- Convalida l'input dell'utente: assicurati che tutti gli input dell'utente siano convalidati per prevenire attacchi di iniezione e altre vulnerabilità di sicurezza. Utilizza librerie e servizi di convalida degli input come Amazon API Gateway e AWS WAFper prevenire i vettori di attacco comuni.
- Usa i ruoli IAM: utilizza i ruoli AWS Identity and Access Management (IAM) per controllare l'accesso alle AWS risorse e assicurarti che solo gli utenti autorizzati abbiano accesso. Segui il principio del privilegio minimo e assicurati che ogni utente disponga solo delle autorizzazioni necessarie per svolgere il proprio ruolo.
- Utilizza una politica in materia di password: configura una politica di password che soddisfi i requisiti di sicurezza, come lunghezza minima, complessità e scadenza. Usa Secrets Manager o AWS Systems Manager Parameter Store per archiviare e gestire le password in modo sicuro.
- Abilita l'autenticazione a più fattori (MFA): abilita l'MFA per tutti gli utenti per fornire un ulteriore livello di sicurezza e ridurre il rischio di accessi non autorizzati. Usa AWS IAM Identity CenterAmazon Cognito per abilitare l'MFA e altri metodi di autenticazione.
- Archivia le informazioni sensibili in modo sicuro: archivia le informazioni sensibili, come password e token di accesso, in modo sicuro utilizzando AWS Key Management Service ()AWS KMS o altri servizi di crittografia.
- Utilizza metodi di autenticazione avanzati: per aumentare la sicurezza del processo di autenticazione, utilizza metodi di autenticazione avanzati, come l'autenticazione biometrica o l'autenticazione a più fattori.
- Monitoraggio di attività sospette: utilizza altri strumenti di monitoraggio per monitorare attività sospette AWS CloudTraile potenziali minacce alla sicurezza. Imposta avvisi automatici per attività insolite e usa Amazon GuardDuty o AWS Security Hub CSPMper rilevare potenziali minacce.
- Rivedi e aggiorna regolarmente le politiche di sicurezza: rivedi e aggiorna regolarmente le politiche e le procedure di sicurezza per assicurarti che soddisfino i requisiti di sicurezza e le migliori pratiche in evoluzione. AWS Config Utilizzatelo per tenere traccia e verificare le modifiche alle politiche e alle procedure di sicurezza.

- Registrazione automatica: non abilitare l'iscrizione automatica a un pool di utenti di Amazon Cognito. Per ulteriori informazioni, consulta Ridurre i rischi di frode nelle iscrizioni degli utenti e di invio di SMS con i pool di utenti di Amazon CognitoAWS (post sul blog).

Per ulteriori best practice, consulta la sezione Best practice di sicurezza per i pool di utenti di Amazon Cognito nella documentazione di Amazon Cognito.

Epiche

Configura le risorse AWS

Operazione	Description	Competenze richieste

Crea un pool di utenti.	1. In un terminale CLI, inserisci il seguente comando per creare un pool di utenti Amazon Cognito:
	`aws cognito-idp create-user-pool \ --pool-name <MyUserPool>`
	2. Inserisci il seguente comando per creare un ID client e un segreto da utilizzare con l'SDK per: JavaScript
	`aws cognito-idp create-user-pool-client \ --user-pool-id <UserPoolID> \ --client-name <MyNewClient> \ --no-generate-secret \ --explicit-auth-flows "USER_PAS`
		Sviluppatore di app, AWS DevOps

Operazione	Description	Competenze richieste
	SWORD_AUTH" "ADMIN_NO_SRP_AUTH"	
	Per ulteriori informazioni e istruzioni su come configura re un pool di utenti in Console di gestione AWS, consulta Introduzione ai pool di utenti e Aggiungere altre funzionalità e opzioni di sicurezza al pool di utenti.	
	Tip Per ridurre i costi, utilizza il piano Essentials o il piano Lite per testare questo modello. Per ulteriori informazioni, consulta i prezzi di Amazon Cognito.	

Operazione	Description	Competenze richieste
Aggiungi un utente al pool di utenti.	Inserisci il seguente comando per creare un utente nel pool di utenti di Amazon Cognito: `aws cognito-idp sign-up \ --client-id <ClientID> \ --username <jane@example.com> \ --password <PASSWORD> \ --user-attributes Name="email",Value="<jane@example.com>" Name="name",Value="<Jane>"`	Sviluppatore di app, AWS DevOps

Operazione	Description	Competenze richieste
Aggiungi il client dell'app al pool di utenti.	1. Passa alla console di Amazon Cognito. 2. Scegli Bacini d'utenza. 3. Scegli il pool di utenti che hai creato in precedenza. 4. In Definisci la tua applicazione, scegli il tipo di applicazione più adatto allo scenario applicativo per il quale desideri creare servizi di autenticazione e autorizzazione. 5. In Assegna un nome all'applicazione, inserisci un nome descrittivo o procedi con il nome predefinito. 6. In Aggiungi un URL di ritorno, inserisci un percorso di reindirizzamento all'applicazione per dopo che gli utenti hanno completato l'autenticazione. 7. Scegli Crea la tua applicazione. Puoi configurare le opzioni avanzate dopo aver creato il client dell'app.	Amministratore di sistema AWS, amministratore AWS, AWS DevOps, sviluppatore di app

Operazione	Description	Competenze richieste
Crea un trigger Lambda per la generazione di pre-token.	1. Apri la pagina Funzioni della console Lambda. 2. Scegli Crea funzione. 3. Scegli Crea da zero. 4. Nel riquadro Informazioni di base, per Nome della funzione, immettere. `myPreTokenGenerationLambdaFunction` 5. Per Runtime, scegliete Node.js 22. 6. Lasciate l'architettura impostata su x86_64. 7. Scegli Crea funzione. 8. Nella scheda Codice, in Codice sorgente, inserisci la funzione Lambda di esempio dalla sezione Risorse aggiuntive di questo modello. Questa funzione Lambda inserisce gli attributi personalizzati nel token ID e nel token di accesso.	AWS DevOps, sviluppatore di app

Operazione	Description	Competenze richieste
Personalizza il flusso di lavoro del pool di utenti.	1. Passa alla console di Amazon Cognito. 2. Scegli Bacini d'utenza. 3. Scegli il pool di utenti che hai creato in precedenza. 4. Nel riquadro di navigazione a sinistra, sotto Autentica zione, scegli Estensioni. 5. Scegli Aggiungi trigger Lambda. 6. Aggiungi o modifica un Trigger di pre-generazione di token. 7. Per il tipo di trigger, scegli Autenticazione. 8. In Autenticazione, scegli Pre token generation trigger. 9. Per Assegna funzione Lambda, scegli. `myPreTokenGenerati onLambdaFunction` 10. Scegli Aggiungi trigger Lambda. 11. Scegli una versione dell'evento Trigger di Funzionalità di base + personalizzazione del token di accesso per le identità utente o Funzional ità di base + personali zzazione del token di	AWS DevOps, sviluppatore di app

Operazione	Description	Competenze richieste
	accesso per le identità di utente e macchina. Questa impostazione aggiorna i parametri di richiesta inviati da Amazon Cognito alla funzione per includere campi per la personali zzazione del token di accesso. Per ulteriori informazioni, consulta Personalizzazione dei flussi di lavoro del pool di utenti con i trigger Lambda nella documentazione di Amazon Cognito.	

Crea l'applicazione Node.js

Operazione	Description	Competenze richieste

Creare l'applicazione Node.js.	1. In un terminale, immettete i seguenti comandi:
	
	`mkdir my-app`
	`cd my-app`
	`npm init -y`
	`npm install --save-de`
	`v typescript`
	`npm i --save-dev`
	`@types/node`
	`npm i amazon-cognito-`
	`identity-js`
	`npx tsc --init`
	`mkdir src`
		Sviluppatore di app

Operazione	Description	Competenze richieste

	`touch src/index.ts`
	2. Modifica il `package.json` file per includere gli script di TypeScript compilazione:
	`"scripts": { "start": "node src/index.js", "build": "tsc" },`	

Implementa la logica di autenticazione.	1. Apri il file `index.ts`. 2. Incolla il Typescript di esempio fornito nella sezione Risorse aggiuntive di questo modello. 3. Salvare e chiudere il file `index.ts`. **Note** È possibile creare il proprio TypeScript file o modificare l'esempio fornito in base alle esigenze del caso d'uso.	Sviluppatore di app

Operazione	Description	Competenze richieste
Configura le variabili di ambiente e il file di configura zione.	In un terminale, inserisci i seguenti comandi per creare le variabili di ambiente: `export USERNAME= "<COGNITO_USER_NAME>" export PASSWORD= "<COGNITO_USER_PAS SWORD>" export USER_POOL _ID="<COGNITO_USER _ID>" export CLIENT_ID ="<COGNITO_CLIENT_ ID>"` **Important** Non codificate segreti né rivelate le vostre credenziali.	Sviluppatore di app
Esegui l'applicazione.	Immettete i seguenti comandi per eseguire l'applicazione e confermare che funzioni: `npm run build npm start`	Sviluppatore di app

Operazione	Description	Competenze richieste
Verificate che gli attributi personalizzati vengano inseriti nei token.	Utilizza le funzionalità di debug del tuo IDE per visualizzare i token di accesso e ID. Conferma che gli attributi personalizzati sono stati aggiunti. Per i token di esempio, consulta la sezione Informazioni aggiuntive di questo modello.	Sviluppatore di app

risoluzione dei problemi

Problema	Soluzione
ID client non valido quando si tenta di autentica re l'utente	Questo errore si verifica in genere quando si utilizza un ID client con un client secret generato. È necessario creare un ID cliente senza un segreto associato. Per ulteriori informazioni, consulta Impostazioni specifiche dell'applicazione con i client dell'app.

Risorse correlate

- Personalizzazione dei flussi di lavoro dei pool di utenti con trigger Lambda (documentazione di Amazon Cognito)
- Trigger Lambda prima della generazione di token (documentazione di Amazon Cognito)
- CognitoIdentityProviderClient(documentazione)AWS SDK per JavaScript
- cognito-idp (documentazione)AWS CLI

Informazioni aggiuntive

File di TypeScript esempio

Il seguente esempio di codice è un TypeScript file che richiama il processo di autenticazione utilizzando un AWS SDK per inviare attributi personalizzati ad Amazon Cognito:

```
import * as AmazonCognitoIdentity from "amazon-cognito-identity-js";

const userPoolId: string = process.env.USER_POOL_ID ?? '';
const clientId: string = process.env.CLIENT_ID ?? '';

const poolData = {
  UserPoolId: userPoolId,
  ClientId: clientId
};
const userPool = new AmazonCognitoIdentity.CognitoUserPool(poolData);

export const loginWithCognitoSDK = function (userName: string, password: string) {
  const authenticationDetails = new AmazonCognitoIdentity.AuthenticationDetails({
    Username: userName,
    Password: password,
    ClientMetadata: {
        customGroup: "MyCustomGroup",
        customApplicationData: "Custom data from a custom application"
    }
  });
  const userData = {
    Username: userName,
    Pool: userPool
  };

  const cognitoUser = new AmazonCognitoIdentity.CognitoUser(userData);

  // Authenticate the user using the authenticationDetails object
  cognitoUser.authenticateUser(authenticationDetails, {
    onSuccess: function (result: any) {},
    onFailure: function (err: any) {},
  });
}
loginWithCognitoSDK(process.env.USERNAME ?? '', process.env.PASSWORD ?? '');
```

L'esempio utilizza il `AuthenticationDetails` modello dell'SDK per JavaScript fornire il nome utente, la password e il. `ClientMetadada` Dopo l'autenticazione in Amazon Cognito, i metadati del client possono essere recuperati dai token di accesso e ID.

Funzione Lambda di esempio

Il seguente esempio di codice è una funzione Lambda collegata al token di pre-generazione di Amazon Cognito. Ti aiuta a personalizzare il token di accesso e il token ID utilizzati da Amazon Cognito. I token vengono passati attraverso le integrazioni tra le tue architetture. Questo esempio include un attributo di reclamo personalizzato chiamato `customApplicationData` e un nome di gruppo personalizzato chiamato: `MyCustomGroup`

```
export const handler = async(event, context, callback) => {
    event.response = {
        claimsOverrideDetails: {
            claimsToAddOrOverride: { customApplicationData:
 event.request.clientMetadata.customApplicationData },
            groupOverrideDetails: { groupsToOverride:
 [event.request.clientMetadata.customGroup] }
        }
    };
    callback(null, event);
};
```

Esempio di token di accesso

È possibile decodificare il token di accesso per visualizzare gli attributi personalizzati che sono stati aggiunti. Di seguito è riportato un esempio di token di accesso:

```
{
  "sub": "6daf331f-4451-48b4-abde-774579299204",
  "cognito:groups": [
    "MyCustomGroup"
  ],
  "iss": "https://cognito-idp.<REGION>.amazonaws.com/<USERPOOL_ID>",
  "client_id": "<YOUR_CLIENT_ID>",
  "origin_jti": "acff7e91-09f9-4fde-8eec-38b0f8c47cdc",
  "event_id": "c5113a9c-1f01-435b-9b73-a5cd3e88514e",
  "token_use": "access",
  "scope": "aws.cognito.signin.user.admin",
  "auth_time": 1677979246,
  "exp": 1677982846,
  "iat": 1677979246,
  "jti": "5c9c2708-a871-4428-bd9b-18ad261bea90",
  "username": "<USER_NAME>"
}
```

Token ID di esempio

È possibile decodificare il token di accesso per visualizzare gli attributi personalizzati che sono stati aggiunti. Di seguito è riportato un esempio di token di accesso:

```
{
  "sub": "6daf331f-4451-48b4-abde-774579299204",
  "cognito:groups": [
    "MyCustomGroup"
  ],
  "iss": "https://cognito-idp.<REGION>.amazonaws.com/<USERPOOL_ID>",
  "cognito:username": "<USER_NAME>",
  "origin_jti": "acff7e91-09f9-4fde-8eec-38b0f8c47cdc",
  "customApplicationData": "Custom data from a custom application",
  "aud": "<YOUR_CLIENT_ID>",
  "event_id": "c5113a9c-1f01-435b-9b73-a5cd3e88514e",
  "token_use": "id",
  "auth_time": 1677979246,
  "exp": 1677982846,
  "iat": 1677979246,
  "jti": "f7ca006b-f25b-44d2-a7a4-6e6423f4201f",
  "email": "<USER_EMAIL>"
}
```

Semplifica la gestione privata dei certificati utilizzando AWS Private CA e AWS RAM

Everett Hinckley e Vivek Goyal, Amazon Web Services

Riepilogo

Puoi utilizzare AWS Private Certificate Authority (AWS Private CA) per emettere certificati privati per l'autenticazione delle risorse interne e la firma del codice informatico. Questo modello fornisce un CloudFormation modello AWS per la rapida implementazione di una gerarchia CA a più livelli e un'esperienza di provisioning coerente. Facoltativamente, puoi utilizzare AWS Resource Access Manager (AWS RAM) per condividere in modo sicuro la CA all'interno delle tue organizzazioni o unità organizzative (OUs) in AWS Organizations e centralizzare la CA utilizzando la RAM AWS per gestire le autorizzazioni. Non è necessaria una CA privata per ogni account, quindi questo approccio consente di risparmiare denaro. Inoltre, puoi utilizzare Amazon Simple Storage Service (Amazon S3) per archiviare l'elenco di revoca dei certificati (CRL) e i log di accesso.

Questa implementazione offre le seguenti caratteristiche e vantaggi:

- Centralizza e semplifica la gestione della gerarchia delle CA private utilizzando AWS Private CA.
- Esporta certificati e chiavi su dispositivi gestiti dai clienti su AWS e in locale.
- Utilizza un CloudFormation modello AWS per una distribuzione rapida e un'esperienza di provisioning coerente.
- Crea una CA root privata con 1, 2, 3 o 4 gerarchie CA subordinate.
- Facoltativamente, utilizza AWS RAM per condividere la CA subordinata dell'entità finale con altri account a livello di organizzazione o unità organizzativa.
- Consente di risparmiare denaro eliminando la necessità di una CA privata in ogni account utilizzando la RAM AWS.
- Crea un bucket S3 opzionale per il CRL.
- Crea un bucket S3 opzionale per i log di accesso CRL.

Prerequisiti e limitazioni

Prerequisiti

Se desideri condividere la CA all'interno di una struttura AWS Organizations, identifica o configura quanto segue:

- Un account di sicurezza per creare e condividere la gerarchia CA.
- Un'unità organizzativa o un account separato per il test.
- Condivisione abilitata all'interno dell'account di gestione AWS Organizations. Per ulteriori informazioni, consulta Abilitare la condivisione delle risorse all'interno di AWS Organizations nella documentazione RAM di AWS.

Limitazioni

- CAs sono risorse regionali. Tutti CAs risiedono in un unico account AWS e in un'unica regione AWS.
- I certificati e le chiavi generati dagli utenti non sono supportati. In questo caso d'uso, si consiglia di personalizzare questa soluzione per utilizzare una CA root esterna.
- Un bucket CRL pubblico non è supportato. Ti consigliamo di mantenere privato il CRL. Se è richiesto l'accesso a Internet al CRL, consulta la sezione sull'utilizzo di Amazon CloudFront per servire CRLs in Abilitazione della funzionalità S3 Block Public Access (BPA) nella documentazione di AWS Private CA.
- Questo modello implementa un approccio a regione singola. Se hai bisogno di un'autorità di certificazione multiregionale, puoi implementare i subordinati in una seconda regione AWS o in locale. Tale complessità non rientra nell'ambito di questo modello, poiché l'implementazione dipende dal caso d'uso specifico, dal volume del carico di lavoro, dalle dipendenze e dai requisiti.

Architecture

Stack tecnologico Target

- CA privata AWS
- AWS RAM
- Simple Storage Service (Amazon S3)
- AWS Organizations
- AWS CloudFormation

Architettura Target

Questo modello offre due opzioni per la condivisione con AWS Organizations:

Opzione 1 – Crea la condivisione a livello di organizzazione. Tutti gli account dell'organizzazione possono emettere i certificati privati utilizzando la CA condivisa, come illustrato nel diagramma seguente.

Opzione 2 – Creare la condivisione a livello di unità organizzativa (OU). Solo gli account dell'unità organizzativa specificata possono emettere i certificati privati utilizzando la CA condivisa. Ad esempio, nel diagramma seguente, se la condivisione viene creata a livello di unità organizzativa Sandbox, sia lo Sviluppatore 1 che lo Sviluppatore 2 possono emettere certificati privati utilizzando la CA condivisa.

Tools (Strumenti)

Servizi AWS

- AWS Private CA — AWS Private Certificate Authority (AWS Private CA) è un servizio CA privato ospitato per l'emissione e la revoca di certificati digitali privati. Ti aiuta a creare gerarchie di CA private, tra cui root e subordinate CAs, senza i costi di investimento e manutenzione legati alla gestione di una CA locale.
- AWS RAM: AWS Resource Access Manager (AWS RAM) ti aiuta a condividere in modo sicuro le tue risorse tra account AWS e all'interno della tua organizzazione o OUs in AWS Organizations. Per ridurre il sovraccarico operativo in un ambiente con più account, puoi creare una risorsa e utilizzare la RAM AWS per condividerla tra più account.
- AWS Organizations — AWS Organizations è un servizio di gestione degli account che consente di consolidare più account AWS in un'organizzazione da creare e gestire centralmente.
- Amazon S3 — Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti. È possibile utilizzare Amazon S3 per memorizzare e recuperare qualsiasi volume di dati, in qualunque momento e da qualunque luogo tramite il Web. Questo modello utilizza Amazon S3 per archiviare l'elenco di revoca dei certificati (CRL) e i log di accesso.
- AWS CloudFormation: AWS ti CloudFormation aiuta a modellare e configurare le tue risorse AWS, effettuarne il provisioning in modo rapido e coerente e gestirle per tutto il loro ciclo di vita. Puoi utilizzare un modello per descrivere le tue risorse e le loro dipendenze e lanciarle e configurarle insieme come uno stack, invece di gestire le risorse singolarmente. Questo modello utilizza AWS CloudFormation per implementare automaticamente una gerarchia CA a più livelli.

Codice

Il codice sorgente di questo pattern è disponibile su GitHub, nel repository AWS Private CA Hierarchy. Il repository include:

- Il CloudFormation modello AWSACMPCA-RootCASubCA.yaml. Puoi utilizzare questo modello per implementare la gerarchia delle CA per questa implementazione.
- File di test per casi d'uso come la richiesta, l'esportazione, la descrizione e l'eliminazione di un certificato.

Per utilizzare questi file, segui le istruzioni nella sezione Epics.

Epiche

Progetta la gerarchia delle CA

Operazione	Description	Competenze richieste
Raccogli informazioni sull'oggetto del certificato.	Raccogli informazioni sull'oggetto del certificato sul proprietario del certificato: nome dell'organizzazione, unità organizzativa, paese, stato, località e nome comune.	Architetto cloud, architetto della sicurezza, ingegnere PKI
Raccogli informazioni opzionali su AWS Organizations.	Se la CA farà parte di una struttura AWS Organizations e desideri condividere la gerarchia della CA all'interno di tale struttura, raccogli il numero dell'account di gestione, l'ID dell'organizzazione e, facoltativamente, l'ID dell'unità organizzativa (se desideri condividere la gerarchia della CA solo con un'unità organizzativa specifica). Inoltre, determina	Architetto del cloud, architetto della sicurezza, ingegnere PKI

Operazione	Description	Competenze richieste
	gli account AWS Organizations o OUs, se ce ne sono, con cui vuoi condividere la CA.	
Progetta la gerarchia CA.	Determina quale account ospiterà il root e il CAs subordinato. Determina quanti livelli subordinati richiede la gerarchia tra i certificati radice e quelli dell'entità finale. Per ulteriori informazioni, consulta Progettazione di una gerarchia di CA nella documentazione di AWS Private CA.	Architetto del cloud, architetto della sicurezza, ingegnere PKI
Determina le convenzioni di denominazione e etichettatura per la gerarchia CA.	Determina i nomi delle risorse AWS: la CA principale e ogni CA subordinata. Determina quali tag devono essere assegnati a ciascuna CA.	Architetto del cloud, architetto della sicurezza, ingegnere PKI

Operazione	Description	Competenze richieste
Determina gli algoritmi di crittografia e firma richiesti.	Determina quanto segue: • Requisiti dell'algoritmo di crittografia dell'organizzazione per le chiavi pubbliche utilizzate dall'autorità di certificazione per l'emissione di un certificato. L'impostazione predefinita è RSA_2048. • L'algoritmo chiave utilizzato dalla CA per la firma dei certificati. L'impostazione predefinita è SHA256 WITHRSA.	Architetto del cloud, architetto della sicurezza, ingegnere PKI
Determina i requisiti di revoca dei certificati per la gerarchia CA.	Se sono necessarie funzionalità di revoca dei certificati, stabilisci una convenzione di denominazione per il bucket S3 che contiene l'elenco di revoca dei certificati (CRL).	Architetto del cloud, architetto della sicurezza, ingegnere PKI
Determina i requisiti di registrazione per la gerarchia CA.	Se sono necessarie funzionalità di registrazione degli accessi, stabilisci una convenzione di denominazione per il bucket S3 che contiene i log di accesso.	Architetto cloud, architetto della sicurezza, ingegnere PKI

Operazione	Description	Competenze richieste
Determina i periodi di scadenza dei certificati.	Determina la data di scadenza del certificato radice (l'impost azione predefinita è 10 anni), dei certificati di entità finale (l'impostazione predefinita è 13 mesi) e dei certifica ti CA subordinati (l'impost azione predefinita è 3 anni). I certificati CA subordina ti devono scadere prima dei certificati CA ai livelli più alti della gerarchia. Per ulteriori informazioni, consulta Managing the private CA lifecycle nella documentazione di AWS Private CA.	Architetto del cloud, architetto della sicurezza, ingegnere PKI

Implementa la gerarchia CA

Operazione	Description	Competenze richieste
Completare i prerequisiti	Completa i passaggi indicati nella sezione Prerequisiti di questo modello.	Amministratore del cloud, ingegneri della sicurezza, ingegneri PKI
Crea ruoli CA per vari personaggi.	1. Determina i tipi di ruoli o utenti di AWS Identity and Access Management (IAM) in AWS IAM Identity Center (successore di AWS Single Sign-On) necessari per amministrare i vari livelli della gerarchia CA, come CAAdmin Root, Subordina	Amministratore del cloud, ingegneri della sicurezza, ingegneri PKI

Operazione	Description	Competenze richieste
	te e. CAAdmin Certificate Consumer 2. Determina la granularità delle politiche necessarie per separare le mansioni. 3. Crea i ruoli o gli utenti IAM richiesti in IAM Identity Center nell'account in cui risiede la gerarchia CA.	
Implementa lo stack. CloudFormation	1. Dal GitHub repository di questo pattern, scarica il template.yaml. AWSPCA-RootCASubCA 2. Distribuisci il modello dalla CloudFormation console AWS o dall'AWS Command Line Interface (AWS CLI). Per ulteriori informazioni, consulta Working with stacks nella documentazione. CloudFormation 3. Completa i parametri nel modello, inclusi il nome dell'organizzazione, il nome dell'unità organizzativa, l'algoritmo chiave, l'algoritmo di firma e altre opzioni.	Amministratore del cloud, ingegneri della sicurezza, ingegneri PKI

Operazione	Description	Competenze richieste
Progetta una soluzione per l'aggiornamento dei certificati utilizzati dalle risorse gestite dagli utenti.	Le risorse dei servizi AWS integrati, come Elastic Load Balancing, aggiornano automaticamente i certifica ti prima della scadenza. Tuttavia, le risorse gestite dagli utenti, come i server Web in esecuzione su istanze Amazon Elastic Compute Cloud EC2 (Amazon), richiedono un altro meccanism o. 1. Determina quali risorse gestite dall'utente richiedon o certificati di entità finale dalla CA privata. 2. Pianifica un processo per ricevere notifiche sulla scadenza delle risorse e dei certificati gestiti dagli utenti. Per esempi di , consulta le sezioni seguenti: • Utilizzo di una regola gestita da AWS Config • Utilizzo di Amazon CloudWatch e Amazon EventBridge 3. Scrivi script personalizzati per aggiornare i certificati sulle risorse gestite dagli utenti e integrali con i servizi AWS per automatiz zare gli aggiornamenti.	Amministratore del cloud, ingegneri della sicurezza, ingegneri PKI

Operazione	Description	Competenze richieste
	Per ulteriori informazioni sui servizi AWS integrati , consulta Servizi integrati con AWS Certificate Manager nella documenta zione ACM.	

Convalida e documenta la gerarchia delle CA

Operazione	Description	Competenze richieste
Convalida la condivisione opzionale di RAM AWS.	Se la gerarchia delle CA è condivisa con altri account in AWS Organizations, accedi a uno di questi account dalla Console di gestione AWS, accedi alla console AWS Private CA e conferma che la CA appena creata sia condivisa con questo account. Sarà visibile solo la CA di livello più basso nella gerarchia, poiché è la CA che genera i certificati dell'entità finale. Ripetere l'operazione per un campione degli account con cui è condivisa la CA.	Amministratore del cloud, ingegneri della sicurezza, ingegneri PKI
Convalida la gerarchia delle CA con test del ciclo di vita dei certificati.	Nell'GitHub archivio relativo a questo modello, individua i test del ciclo di vita. Esegui i test dalla CLI di AWS per richieder e un certificato, esportare un certificato, descrivere un	Amministratore del cloud, ingegneri della sicurezza, ingegneri PKI

Operazione	Description	Competenze richieste
	certificato ed eliminare un certificato.	
Importa la catena di certificati in Trust Stores.	Affinché i browser e le altre applicazioni considerino attendibile un certificato, l'emittente del certificato deve essere incluso nell'archivio attendibile del browser, che è un elenco di certificati attendibili CAs. Aggiungi la catena di certificati per la nuova gerarchia CA all'archivio attendibile del browser e dell'applicazione. Verifica che i certificati dell'entità finale siano attendibili.	Amministratore del cloud, ingegneri della sicurezza, ingegneri PKI
Crea un runbook per documentare la gerarchia delle CA.	Crea un documento di runbook per descrivere l'architettura della gerarchia delle CA, la struttura degli account che può richiedere certificati di entità finale, il processo di compilazione e le attività di gestione di base come l'emissione di certificati di entità finale (a meno che non si desideri consentire il self-service da parte degli account secondari), l'utilizzo e il monitoraggio.	Amministratore del cloud, ingegneri della sicurezza, ingegneri PKI

Risorse correlate

- Progettazione di una gerarchia CA (documentazione di AWS Private CA)
- Creazione di una CA privata (documentazione AWS Private CA)
- Come usare la RAM AWS per condividere il tuo cross-account AWS Private CA (post sul blog AWS)
- Best practice di AWS Private CA (post sul blog AWS)
- Abilita la condivisione delle risorse all'interno di AWS Organizations (documentazione RAM AWS)
- Gestione del ciclo di vita della CA privata (documentazione AWS Private CA)
- acm-certificate-expiration-check per AWS Config (documentazione AWS Config)
- AWS Certificate Manager ora fornisce il monitoraggio della scadenza dei certificati tramite Amazon CloudWatch (annuncio AWS)
- Servizi integrati con AWS Certificate Manager (documentazione ACM)

Informazioni aggiuntive

Quando esporti certificati, usa una passphrase crittograficamente sicura e in linea con la strategia di prevenzione della perdita di dati della tua organizzazione.

Semplifica la gestione EC2 della conformità di Amazon con agenti Amazon Bedrock e AWS Config

Anand Bukkapatnam Tirumala, Amazon Web Services

Riepilogo

Questo modello descrive come integrare Amazon Bedrock con AWS Config regole per facilitare la gestione della conformità per le istanze di Amazon Elastic Compute Cloud EC2 (Amazon). L'approccio utilizza funzionalità di intelligenza artificiale generativa avanzate per fornire consigli personalizzati in linea con il AWS Well-Architected Framework, per garantire una selezione ottimale del tipo di istanza e l'efficienza del sistema. Le caratteristiche principali di questo modello includono:

- Monitoraggio automatizzato della conformità: AWS Config le regole valutano continuamente le EC2 istanze in base a criteri predefiniti per i tipi di istanze desiderati.
- Consigli basati sull'intelligenza artificiale: i modelli di intelligenza artificiale generativa di Amazon Bedrock analizzano i modelli di infrastruttura. Questi modelli forniscono suggerimenti intelligenti per miglioramenti basati sulle migliori pratiche descritte nel AWS Well-Architected Framework.
- Correzione: i gruppi di azione di Amazon Bedrock abilitano procedure di riparazione automatizzate per risolvere rapidamente le istanze non conformi e ridurre al minimo le potenziali inefficienze in termini di prestazioni o costi.
- Scalabilità e adattabilità: la soluzione è progettata per adattarsi alla tua infrastruttura e adattarsi alle esigenze in evoluzione dell'architettura cloud.
- Raccomandazioni di sicurezza avanzate: la conformità ai principi AWS Well-Architected contribuisce a migliorare il livello di sicurezza e le prestazioni del sistema.

È possibile utilizzare questo modello come modello per implementare la propria infrastruttura generativa basata sull'intelligenza artificiale in più ambienti con modifiche minime, utilizzando le procedure necessarie. DevOps

Prerequisiti e limitazioni

Prerequisiti

- Un attivo. Account AWS

- Un ruolo AWS Identity and Access Management (IAM) con autorizzazioni per creare e gestire risorse nei bucket e nelle AWS Config funzioni di Amazon Simple Storage Service (Amazon S3), Amazon Bedrock AWS Lambda , IAM, Amazon Logs e Amazon. CloudWatch EC2
- Un' EC2 istanza da contrassegnare come non conforme. Non utilizzate il `t2.small` tipo per questa istanza.
- Modelli Amazon Titan Text Embeddings V2 e Anthropic Claude 3 Haiku abilitati nel tuo. Account AWS Per abilitare l'accesso ai modelli per il Regione AWS luogo in cui stai distribuendo la soluzione, consulta Aggiungere o rimuovere l'accesso ai modelli Amazon Bedrock Foundation nella documentazione di Amazon Bedrock.
- Terraform, installato e configurato.
- La AWS Command Line Interface (AWS CLI) v2 installata e configurata nell'ambiente di distribuzione.
- Revisione completata della politica di Amazon Responsible AI.

Limitazioni

- Alcune Servizi AWS non sono disponibili in tutte Regioni AWS. Per la disponibilità per regione, vedi Servizi AWS per regione. Per endpoint specifici, consulta Endpoints and quotas del servizio e scegli il link relativo al servizio.
- Questa soluzione è stata testata utilizzando i modelli Amazon Titan Text Embeddings V2 e Claude 3 Haiku. Se preferisci utilizzare altri modelli, puoi personalizzare il codice Terraform, che è parametrizzato per facilitare le modifiche.
- Questa soluzione non include una funzionalità di cronologia chat e la chat non viene archiviata.

Architecture

Il diagramma seguente mostra i componenti del flusso di lavoro e dell'architettura per questo modello.

Il flusso di lavoro è composto dai seguenti passaggi:

1. L'utente interagisce con il modello tramite la console di chat Amazon Bedrock. L'utente pone domande come:
   - `What can you help me with?`
   - `List non-complaint resources`

- `Suggest security best practices`

2. Se il modello è preaddestrato, risponde alle richieste direttamente in base alle conoscenze esistenti. In caso contrario, il prompt viene inviato al gruppo di azione Amazon Bedrock.
3. Il gruppo di azione raggiunge gli endpoint VPC utilizzando AWS PrivateLinkper una comunicazione di servizio sicura.
4. La richiesta raggiunge la funzione Lambda tramite gli endpoint VPC per i servizi Amazon Bedrock.
5. La funzione Lambda è il motore di esecuzione principale. In base alla richiesta, la funzione chiama l'API per eseguire azioni su. Servizi AWS Gestisce inoltre il routing e l'esecuzione delle operazioni.
6. La funzione Lambda chiama AWS Config per determinare le risorse non conformi (l' EC2 istanza non conforme che hai creato come prerequisito).
7. AWS Config contrassegna la risorsa che non presenta reclami. Questo modello utilizza la AWS Config desired-instance-typeregola per trovare la dimensione ideale EC2 dell'istanza.
8. AWS Config richiede all'utente di sospendere o ripristinare l'istanza e interviene di conseguenza sull'istanza. EC2 Amazon Bedrock comprende questo payload di reso.
9. L'utente riceve una risposta sulla console di chat di Amazon Bedrock.

Automazione e scalabilità

Questa soluzione utilizza Terraform come strumento Infrastructure as Code (IaC) per consentire una facile implementazione Account AWS e funzionare come utilità autonoma su più account. Questo approccio semplifica la gestione e migliora la coerenza nelle implementazioni.

Tools (Strumenti)

Servizi AWS

- AWS Configconsente di valutare, controllare e valutare le configurazioni delle AWS risorse per quanto riguarda la conformità e le impostazioni desiderate.
- Amazon Bedrock è un servizio di intelligenza artificiale completamente gestito che fornisce l'accesso a molti modelli di base ad alte prestazioni tramite un'API unificata.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Lambda è un servizio di calcolo che consente di eseguire il codice senza gestire i server o effettuarne il provisioning. Esegue il codice solo quando necessario e si ridimensiona automaticamente, quindi paghi solo per il tempo di elaborazione che utilizzi.

Altri strumenti

- Git è un sistema di controllo delle versioni distribuito e open source.
- Terraform è uno strumento Infrastructure as Code (IaC) HashiCorp che ti aiuta a creare e gestire risorse cloud e locali.

Deposito di codici

Il codice per questo pattern è disponibile nel repository GitHub sample-awsconfig-bedrock-compliance-manager.

Best practice

- Segui il principio del privilegio minimo e concedi le autorizzazioni minime richieste per eseguire un'attività. Per ulteriori informazioni, consulta le best practice e i casi d'uso relativi alla concessione dei privilegi minimi e alla sicurezza nella documentazione IAM.
- Monitora regolarmente i log di esecuzione di Lambda. Per ulteriori informazioni, consulta Monitoraggio, debug e risoluzione dei problemi delle funzioni Lambda e Best practice per l'utilizzo delle funzioni AWS Lambda nella documentazione di Lambda.

Epiche

Implementazione della soluzione

Operazione	Description	Competenze richieste
Clonare il repository.	Per clonare il repository per questo pattern, utilizzate il seguente comando: `git clone "git@github.com:aws-samples/sample-awsconfig-bedrock-compliance-manager.git"`	AWS DevOps, responsabile di sviluppo, DevOps ingegnere, amministratore cloud
Modifica le variabili di ambiente.	Nella directory principale del repository clonato sul	Amministratore di sistema AWS, AWS DevOps, DevOps

Operazione	Description	Competenze richieste
	computer locale, modifica il file. `terraform.tfvars` Esamina i segnaposti contrassegnati con e modifical i in `[XXXXX]` base al tuo ambiente.	ingegnere, amministratore AWS

Crea l'infrastruttura.	1. Per creare l'infrastruttura per questa soluzione, esegui i seguenti comandi:
	`terraform init`
	`terraform plan`
	2. Esamina il piano di esecuzione, quindi esegui il comando:
	`terraform apply --auto-approve`
		AWS DevOps, DevOps ingegnere, amministratore di sistema AWS, amministratore cloud

Usa l'agente

Operazione	Description	Competenze richieste
Chatta con l'agente.	L'implementazione della soluzione nella fase precedent e prevede l'implementazione di `security-bot-agent` un agente Amazon Bedrock con una console di chat. Per utilizzare l'agente: 1. Accedi a Console di gestione AWS con un'identi	AWS DevOps, DevOps ingegnere, amministratore di sistema AWS, amministratore cloud

Operazione	Description	Competenze richieste

	tà IAM che dispone delle autorizzazioni per utilizzar e la console Amazon Bedrock. Quindi, apri la console Amazon Bedrock.
	2. Nel riquadro di navigazio ne, scegli Build, Agents. Seleziona security-bot-agent.
	3. Inizia a conversare con l'agente. Esempio:
	`List non-complaint EC2 instances`
	e:
	`Replace these non-complaint EC2 instances with compliant instances`	

Eseguire la pulizia delle risorse

Operazione	Description	Competenze richieste
Eliminare l'infrastruttura e le risorse.	Una volta completato il lavoro con questa soluzione, puoi eliminare l'infrastruttura creata da questo schema eseguendo il comando: `terraform destroy --auto-approve`	AWS DevOps, DevOps ingegnere, amministratore di sistema AWS, amministratore cloud

risoluzione dei problemi

Problema	Soluzione
Problemi di comportamento degli agenti	Per informazioni sulla risoluzione dei problemi, consulta Test e risoluzione dei problemi del comportamento degli agenti nella documentazione di Amazon Bedrock.
AWS Lambda problemi di rete	Per ulteriori informazioni, consulta Risoluzione dei problemi di rete in Lambda nella documentazione di Lambda.
autorizzazioni IAM	Per ulteriori informazioni, consulta Risoluzione dei problemi di IAM nella documentazione IAM.

Risorse correlate

- Agenti Amazon Bedrock
- Utilizza i gruppi di azioni per definire le azioni che il tuo agente deve eseguire (documentazione Amazon Bedrock)
- desired-instance-type regola (AWS Config documentazione)
- Come AWS Config funziona (AWS Config documentazione)

Aggiorna le credenziali dell'interfaccia a riga di comando AWS da AWS IAM Identity Center utilizzando PowerShell

Chad Miles e Andy Bowen, Amazon Web Services

Riepilogo

Se desideri utilizzare le credenziali AWS IAM Identity Center (successore di AWS Single Sign-On) con AWS Command Line Interface (AWS CLI), AWS SDKs o AWS Cloud Development Kit (AWS CDK), in genere devi copiare e incollare le credenziali dalla console IAM Identity Center nell'interfaccia a riga di comando. Questo processo può richiedere molto tempo e deve essere ripetuto per ogni account che richiede l'accesso.

Una soluzione comune consiste nell'utilizzare il comando AWS CLI`aws sso configure`. Questo comando aggiunge un profilo abilitato per IAM Identity Center alla tua CLI AWS o all'SDK AWS. Tuttavia, lo svantaggio di questa soluzione è che devi eseguire il comando `aws sso login` per ogni profilo o account AWS CLI configurato in questo modo.

Come soluzione alternativa, questo modello descrive come utilizzare i profili denominati dell'interfaccia a riga di comando AWS e gli strumenti AWS per PowerShell archiviare e aggiornare contemporaneamente le credenziali per più account da una singola istanza di IAM Identity Center. Lo script archivia inoltre i dati della sessione IAM Identity Center in memoria per aggiornare le credenziali senza accedere nuovamente a IAM Identity Center.

Prerequisiti e limitazioni

Prerequisiti

- PowerShell, installato e configurato. Per ulteriori informazioni, vedere Installazione PowerShell (documentazione Microsoft).
- Strumenti AWS per PowerShell, installati e configurati. Per motivi di prestazioni, consigliamo vivamente di installare la versione modulare di AWS Tools for PowerShell, chiamata. `AWS.Tools` Ogni servizio AWS è supportato da un piccolo modulo individuale. Nel PowerShell prompt, inserisci i seguenti comandi per installare i moduli necessari per questo modello: `AWS.Tools.InstallerSSO`, e`SSOIDC`.

```
Install-Module AWS.Tools.Installer
Install-AWSToolsModule SSO, SSOOIDC
```

Per ulteriori informazioni, consulta Installare AWS.Tools su Windows o Installare AWS.Tools su Linux o macOS.

- L'AWS CLI o l'SDK AWS devono essere precedentemente configurati con credenziali di lavoro eseguendo una delle seguenti operazioni:
  - Usa il comando AWS CLI`aws configure`. Per ulteriori informazioni, consulta Quick configuration (documentazione AWS CLI).
  - Configura AWS CLI o AWS CDK per ottenere l'accesso temporaneo tramite un ruolo IAM. Per ulteriori informazioni, consulta Ottenere le credenziali del ruolo IAM per l'accesso alla CLI (documentazione IAM Identity Center).

Limitazioni

- Questo script non può essere utilizzato in una pipeline o in una soluzione completamente automatizzata. Quando si distribuisce questo script, è necessario autorizzare manualmente l'accesso da IAM Identity Center. Lo script continua quindi automaticamente.

Versioni del prodotto

- Per tutti i sistemi operativi, si consiglia di utilizzare la PowerShell versione 7.0 o successiva.

Architecture

Puoi utilizzare lo script in questo modello per aggiornare contemporaneamente più credenziali IAM Identity Center e creare un file di credenziali da utilizzare con AWS CLI, AWS o AWS CDK. SDKs

Tools (Strumenti)

Servizi AWS

- AWS Command Line Interface (AWS CLI) è uno strumento open source che ti aiuta a interagire con i servizi AWS tramite comandi nella tua shell a riga di comando.
- AWS IAM Identity Center ti aiuta a gestire centralmente l'accesso Single Sign-On (SSO) a tutti i tuoi account AWS e le tue applicazioni cloud.
- AWS Tools for PowerShell è un set di PowerShell moduli che ti aiutano a creare script di operazioni sulle tue risorse AWS dalla PowerShell riga di comando.

Altri strumenti

- PowerShellè un programma di gestione dell'automazione e della configurazione di Microsoft che funziona su Windows, Linux e macOS.

Best practice

Conserva una copia di questo script per ogni istanza di IAM Identity Center. L'utilizzo di uno script per più istanze non è supportato.

Epiche

Esegui lo script SSO

Operazione	Description	Competenze richieste
Personalizza lo script SSO.	1. Copia lo script SSO nella sezione Informazioni aggiuntive. 2. Nella `Param` sezione, per il tuo ambiente AWS, definisci i valori per le seguenti variabili: • `DefaultRoleName` — Il ruolo o il set di autorizza zioni IAM da utilizzare di default. • `Region`— La regione AWS in cui è distribui to IAM Identity Center. Per un elenco completo delle regioni e dei relativi codici, consulta Endpoint regionali. • `StartUrl`— L'URL utilizzato per accedere alla pagina di accesso	Amministratore del cloud

Operazione	Description	Competenze richieste
	di IAM Identity Center. Utilizza lo stesso formato del valore di esempio nello script. • `EnvironmentName` — Un nome breve per fare riferimento a questa copia dello script, da utilizzar e quando si eseguono più copie di script nella stessa sessione. 3. Alla riga 10, che recita# `Add your Account Information` , modificat e i seguenti valori nelle tabelle hash in modo che rispecchino il vostro ambiente: • `Profile`— Il nome del profilo AWS CLI in cui archiviare le credenziali temporanee. • `AccountId` — L'ID dell'account AWS per il quale stai recuperando le credenziali. • `RoleName`— Il nome del ruolo o del set di autorizzazioni di IAM Identity Center che desideri utilizzare. Puoi lasciare questo valore come `$DefaultR oleName` se volessi	

Operazione	Description	Competenze richieste
	utilizzare lo stesso ruolo definito nella `Param` sezione. Ogni riga nella tabella hash deve terminare con una virgola tranne l'ultima.	

Esegui lo script SSO.	Si consiglia di eseguire lo script personalizzato nella PowerShell shell con il seguente comando.
	
	`./Set-AwsCliSsoCredentials.ps1`
	
	In alternativa, è possibile eseguire lo script da un'altra shell immettendo il seguente comando.
	
	`pwsh Set-AwsCliSsoCredentials.ps1`
		Amministratore cloud

risoluzione dei problemi

Problema	Soluzione
Errore `No Access`	Il ruolo IAM che stai utilizzando non dispone delle autorizzazioni per accedere al ruolo o al set di autorizzazioni definito in un `RoleName` parametro. Aggiorna le autorizzazioni per il ruolo che stai utilizzando o definisci un ruolo o un set di autorizzazioni diverso nello script.

Risorse correlate

- Dove vengono archiviate le impostazioni di configurazione? (documentazione dell'interfaccia a riga di comando di AWS)
- Configurazione dell'interfaccia a riga di comando di AWS per l'utilizzo di AWS IAM Identity Center (documentazione AWS CLI)
- Utilizzo di profili denominati (documentazione dell'interfaccia a riga di comando di AWS)

Informazioni aggiuntive

script SSO

Nello script seguente, sostituisci i segnaposto tra parentesi angolari (**<>**) con le tue informazioni e rimuovi le parentesi angolari.

```
Set-AwsCliSsoCredentials.ps1
Param(
    $DefaultRoleName = '<AWSAdministratorAccess>',
    $Region          = '<us-west-2>',
    $StartUrl        = "<https://d-12345abcde.awsapps.com/start/>",
    $EnvironmentName = "<CompanyName>"
)
Try {$SsoAwsAccounts = (Get-Variable -name "$($EnvironmentName)SsoAwsAccounts" -Scope
 Global -ErrorAction 'SilentlyContinue').Value.Clone()}
Catch {$SsoAwsAccounts = $False}
if (-not $SsoAwsAccounts) { $SsoAwsAccounts = @(
# Add your account information in the list of hash tables below, expand as necessary,
 and do not forget the commas
    @{Profile = "<Account1>"      ; AccountId = "<012345678901 >"; RoleName =
 $DefaultRoleName },
    @{Profile = "<Account2>"      ; AccountId = "<123456789012>"; RoleName =
 "<AWSReadOnlyAccess>" }
)}
$ErrorActionPreference = "Stop"
if (-not (Test-Path ~\.aws))      { New-Item ~\.aws -type Directory }
if (-not (Test-Path ~\.aws\credentials)) { New-Item ~\.aws\credentials -type File }
$CredentialFile = Resolve-Path ~\.aws\credentials
$PsuedoCreds    = @{AccessKey =
 'AKAEXAMPLE123ACCESS';SecretKey='PsuedoS3cret4cceSSKey123PsuedoS3cretKey'} # Pseudo
 Creds, do not edit.
```

```
Try {$SSOTokenExpire = (Get-Variable -Scope Global -Name
 "$($EnvironmentName)SSOTokenExpire" -ErrorAction 'SilentlyContinue').Value} Catch
 {$SSOTokenExpire = $False}
Try {$SSOToken       = (Get-Variable -Scope Global -Name "$($EnvironmentName)SSOToken"
 -ErrorAction 'SilentlyContinue').Value }      Catch {$SSOToken       = $False}
if ( $SSOTokenExpire -lt (Get-Date) ) {
    $SSOToken = $Null
    $Client   = Register-SSOOIDCClient -ClientName cli-sso-client -ClientType public -
Region $Region @PsuedoCreds
    $Device   = $Client | Start-SSOOIDCDeviceAuthorization -StartUrl $StartUrl -Region
 $Region @PsuedoCreds
    Write-Host "A Browser window should open. Please login there and click ALLOW." -
NoNewline
    Start-Process $Device.VerificationUriComplete
    While (-Not $SSOToken){
        Try {$SSOToken = $Client | New-SSOOIDCToken -DeviceCode $Device.DeviceCode -
GrantType "urn:ietf:params:oauth:grant-type:device_code" -Region $Region @PsuedoCreds}
        Catch {If ($_.Exception.Message -notlike "*AuthorizationPendingException*")
{Write-Error $_.Exception} ; Start-Sleep 1}
    }
    $SSOTokenExpire = (Get-Date).AddSeconds($SSOToken.ExpiresIn)
    Set-Variable -Name "$($EnvironmentName)SSOToken" -Value $SSOToken -Scope Global
    Set-Variable -Name "$($EnvironmentName)SSOTokenExpire" -Value $SSOTokenExpire -
Scope Global
}
$CredsTime     = $SSOTokenExpire - (Get-Date)
$CredsTimeText = ('{0:D2}:{1:D2}:{2:D2} left on SSO Token' -f $CredsTime.Hours,
 $CredsTime.Minutes, $CredsTime.Seconds).TrimStart("0 :")
for ($i = 0; $i -lt $SsoAwsAccounts.Count; $i++) {
    if (([DateTimeOffset]::FromUnixTimeSeconds($SsoAwsAccounts[$i].CredsExpiration /
 1000)).DateTime -lt (Get-Date).ToUniversalTime()) {
        Write-host "`r
    `rRegistering Profile $($SsoAwsAccounts[$i].Profile)" -NoNewline
        $TempCreds = $SSOToken | Get-SSORoleCredential -AccountId
 $SsoAwsAccounts[$i].AccountId -RoleName $SsoAwsAccounts[$i].RoleName -Region $Region
 @PsuedoCreds
        [PSCustomObject]@{AccessKey = $TempCreds.AccessKeyId; SecretKey =
 $TempCreds.SecretAccessKey; SessionToken = $TempCreds.SessionToken
        } | Set-AWSCredential -StoreAs $SsoAwsAccounts[$i].Profile -ProfileLocation
 $CredentialFile
        $SsoAwsAccounts[$i].CredsExpiration = $TempCreds.Expiration
    }
}
```

```
Set-Variable -name "$($EnvironmentName)SsoAwsAccounts" -Value $SsoAwsAccounts.Clone() -
Scope Global
Write-Host "`r$($SsoAwsAccounts.Profile) Profiles registered, $CredsTimeText"
```

Usa Network Firewall per acquisire i nomi di dominio DNS dall'Indicazione del nome del server per il traffico in uscita

Kirankumar Chandrashekar, Amazon Web Services

Riepilogo

Questo modello mostra come utilizzare AWS Network Firewall per raccogliere i nomi di dominio DNS forniti dalla Server Name Indication (SNI) nell'intestazione HTTPS del traffico di rete in uscita. Network Firewall è un servizio gestito che semplifica l'implementazione di protezioni di rete critiche per Amazon Virtual Private Cloud (Amazon VPC), inclusa la possibilità di proteggere il traffico in uscita con un firewall che blocca i pacchetti che non soddisfano determinati requisiti di sicurezza. La protezione del traffico in uscita verso nomi di dominio DNS specifici si chiama filtro in uscita, che consiste nel monitorare e potenzialmente limitare il flusso di informazioni in uscita da una rete all'altra.

Dopo aver acquisito i dati SNI che passano attraverso Network Firewall, puoi utilizzare Amazon CloudWatch Logs e AWS Lambda per pubblicare i dati su un argomento Amazon Simple Notification Service (Amazon SNS) che genera notifiche e-mail. Le notifiche e-mail includono il nome del server e altre informazioni SNI pertinenti. Inoltre, è possibile utilizzare l'output di questo pattern per consentire o limitare il traffico in uscita in base al nome di dominio nell'SNI utilizzando le regole del firewall. Per ulteriori informazioni, consulta Lavorare con gruppi di regole stateful in AWS Network Firewall nella documentazione di Network Firewall.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- AWS Command Line Interface (AWS CLI) versione 2, installata e configurata su Linux, macOS o Windows.
- Network Firewall, configurato e configurato in Amazon VPC e utilizzato per ispezionare il traffico in uscita. È possibile configurare Network Firewall per utilizzare una delle seguenti configurazioni VPC:
  - Architettura semplice a zona singola con un gateway Internet
  - Architettura multizona con un gateway Internet

- Architettura con un gateway Internet e un gateway NAT

Architecture

Il diagramma seguente mostra come utilizzare Network Firewall per raccogliere dati SNI dal traffico di rete in uscita e quindi pubblicare tali dati su un argomento SNS utilizzando CloudWatch Logs e Lambda.

Il diagramma mostra il flusso di lavoro seguente:

1. Network Firewall raccoglie i nomi di dominio dai dati SNI nell'intestazione HTTPS del traffico di rete in uscita.
2. CloudWatch Logs monitora i dati SNI e richiama una funzione Lambda ogni volta che il traffico di rete in uscita passa attraverso Network Firewall.
3. La funzione Lambda legge i dati SNI acquisiti da CloudWatch Logs e quindi li pubblica su un argomento SNS.
4. L'argomento SNS ti invia una notifica e-mail che include i dati SNI.

Automazione e scalabilità

- Puoi usare AWS CloudFormation per creare questo modello utilizzando l'infrastruttura come codice.

Stack tecnologico

- CloudWatch Registri Amazon
- Amazon SNS
- Amazon VPC
- AWS Lambda
- AWS Network Firewall

Tools (Strumenti)

Servizi AWS

- Amazon CloudWatch Logs: puoi utilizzare Amazon CloudWatch Logs per monitorare, archiviare e accedere ai tuoi file di log da istanze Amazon Elastic Compute Cloud (Amazon EC2), CloudTrail AWS, Amazon Route 53 e altre fonti.
- Amazon SNS — Amazon Simple Notification Service (Amazon SNS) è un servizio gestito che fornisce il recapito dei messaggi dagli editori agli abbonati (noti anche come produttori e consumatori).
- Amazon VPC: Amazon Virtual Private Cloud (Amazon VPC) fornisce una sezione logicamente isolata del cloud AWS in cui è possibile avviare le risorse AWS in una rete virtuale definita dall'utente. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.
- AWS Lambda: AWS Lambda è un servizio di elaborazione che consente di eseguire codice senza effettuare il provisioning o la gestione di server.
- AWS Network Firewall: AWS Network Firewall è un servizio gestito che semplifica l'implementazione delle protezioni di rete essenziali per tutto il tuo Amazon. VPCs

Epiche

Creare un gruppo di CloudWatch log per Network Firewall

Operazione	Description	Competenze richieste
Crea un gruppo di CloudWatch log.	1. Accedi alla console di gestione AWS e apri la console CloudWatch . 2. Nel pannello di navigazione, selezionare Log groups (Gruppi di log). 3. Selezionare Actions (Operazioni) e scegliere Create log group (Crea gruppo di log). 4. Immettere un nome per il gruppo di log, quindi selezionare Create log group (Crea gruppo di log).	Amministratore cloud

Operazione	Description	Competenze richieste
	Per ulteriori informazioni, consulta Lavorare con gruppi di log e flussi di log nella CloudWatch documentazione.	

Crea un argomento e un abbonamento SNS

Operazione	Description	Competenze richieste
Creare un argomento SNS.	Per creare un argomento SNS, segui le istruzioni nella documentazione di Amazon SNS.	Amministratore cloud
Sottoscrivi un endpoint all'argomento SNS.	Per iscrivere un indirizzo e-mail come endpoint all'argomento SNS che hai creato, segui le istruzioni nella documentazione di Amazon SNS. Per Protocollo, scegli Email/email-JSON. **Note** Puoi anche scegliere un endpoint diverso in base alle tue esigenze.	Amministratore cloud

Configurare la registrazione in Network Firewall

Operazione	Description	Competenze richieste
Abilita la registrazione del firewall.	1. Accedi alla Console di gestione AWS e apri la console Amazon VPC. 2. Nel pannello di navigazione, in NETWORK FIREWALL, scegli Firewalls. 3. Nella sezione Firewall, scegli il firewall in cui desideri acquisire il nome del server dal SNI per il traffico in uscita. 4. Scegli la scheda Dettagli del firewall, quindi scegli Modifica nella sezione Registrazione. 5. Per Tipo di registro, seleziona Avviso. Per Destinazione di registro per gli avvisi, seleziona il gruppo di CloudWatch log. 6. Per Gruppo di CloudWatch log, cerca e scegli il gruppo di log che hai creato in precedenza, quindi scegli Salva. Per ulteriori informazioni sull'utilizzo di CloudWatc h Logs come destinazione di log per Network Firewall, consulta Amazon CloudWatch	Amministratore cloud

Operazione	Description	Competenze richieste
	Logs nella documentazione di Network Firewall.	

Imposta una regola statica in Network Firewall

Operazione	Description	Competenze richieste
Crea una regola statica.	1. Accedi alla Console di gestione AWS e apri la console Amazon VPC. 2. Nel riquadro di navigazione, in NETWORK FIREWALL, scegli Network Firewall Rule Groups. 3. Scegli Crea gruppo di regole Network Firewall. 4. Nella pagina Crea gruppo di regole Network Firewall, per il tipo di gruppo di regole, scegli Stateful rule group. Nota: per ulteriori informazioni, consulta Working with stateful rule group in AWS Network Firewall. 5. Nella sezione Stateful rule group, inserisci un nome e una descrizione per il gruppo di regole. 6. Per Capacità, imposta la capacità massima che desideri consentire per il gruppo di regole stateful (fino a un massimo di	Amministratore cloud

Operazione	Description	Competenze richieste
	30.000). Nota: non puoi modificare questa impostazione dopo aver creato il gruppo di regole. Per informazioni su come calcolare la capacità, consulta Impostazione della capacità del gruppo di regole in AWS Network Firewall. Per informazioni sull'impostazione massima, consulta le quote di AWS Network Firewall. 7. Per le opzioni del gruppo di regole Stateful, seleziona 5-tuple. 8. Nella sezione Stateful rule order, scegli Default. 9. Nella sezione Variabili delle regole, mantieni i valori predefiniti. 10. Nella sezione Aggiungi regola, scegli TLS per Protocollo. Per Origine, scegli Qualsiasi. Per Porta di origine, scegli Qualsiasi porta. Per Destinazione, scegli Qualsiasi. Per Porta di destinazione, scegli Qualsiasi porta. Per Direzione del traffico, scegli Avanti. In Azione, scegli Avviso. Scegli Aggiungi regola.	

Operazione	Description	Competenze richieste
	11. Scegli Crea gruppo di regole stateful.	
Associa la regola stateful a Network Firewall.	1. Accedi alla Console di gestione AWS e apri la console Amazon VPC. 2. Nel pannello di navigazione, in NETWORK FIREWALL, scegli Firewalls. 3. Scegli il firewall in cui desideri acquisire il nome del server dal SNI per il traffico in uscita. 4. Nella sezione Gruppi di regole con stato, scegli Azioni, quindi scegli Aggiungi gruppi di regole con stato non gestiti. 5. Nella pagina Aggiungi gruppi di regole con stato non gestiti, seleziona il gruppo di regole con stato creato in precedenza, quindi scegli Aggiungi gruppo di regole con stato.	Amministratore cloud

Crea una funzione Lambda per leggere i log

Operazione	Description	Competenze richieste
Crea il codice per la funzione Lambda.	In un ambiente di sviluppo integrato (IDE) in grado di leggere l'evento CloudWatch Logs di Network Firewall per	Sviluppatore di app

Operazione	Description	Competenze richieste

	il traffico in uscita, incolla il seguente codice Python 3 e `<SNS-topic-ARN>` sostituiscilo con il tuo valore:
	
	`import json`
	`import gzip`
	`import base64`
	`import boto3`
	`sns_client = boto3.client('sns')`
	`def lambda_handler(event, context):`
	`  decoded_event = json.loads(gzip.decompress(base64.b64decode(event['awslogs']['data'])))`
	`  body = '''`
	`  {filtermatch}`
	`  '''.format(`
	`  loggroup=decoded_event['logGroup'],`
	`  logstream=decoded_event['logStream'],`
	`  filtermatch=decoded_event['logEvents'][0]['message'],`
	`  )`
	`  print(body)`
	`  filterMatch = json.loads(body)`
	`  data = []`
	`  if 'http' in filterMatch['event']:`
	`  data.append(filterMatch['ev`	

Operazione	Description	Competenze richieste
	ent']['http']['hos tname']) elif 'tls' in filterMatch['event']: data.appe nd(filterMatch['ev ent']['tls']['sni']) result = 'Domain accessed ' + 1*' ' + (data[0]) + 1*' ' 'via AWS Network Firewall ' + 1*' ' + (filterMa tch['firewall_name']) print(result) message = {'ServerN ame': result} send_to_sns = sns_client.publish(TargetArn=<SNS-topic-ARN>, #Replace with the SNS topic ARN Message=j son.dumps({'default': json.dumps(message), 'sms': json.dump s(message), 'email': json.dump s(message)}), Subject='Server Name passed through the Network Firewall', MessageSt ructure='json') Questo esempio di codice analizza il contenuto dei CloudWatch Logs e acquisisc	

Operazione	Description	Competenze richieste
	e il nome del server fornito da SNI nell'intestazione HTTPS.	
Creazione della funzione Lambda	Per creare la funzione Lambda, segui le istruzioni nella documentazione di Lambda e scegli Python 3.9 for Runtime.	Amministratore cloud
Aggiungi il codice alla funzione Lambda.	Per aggiungere il codice Python alla funzione Lambda creata in precedenza, segui le istruzioni nella documentazione di Lambda.	Amministratore cloud

Operazione	Description	Competenze richieste
Aggiungi CloudWatch i log come trigger alla funzione Lambda.	1. Accedi alla Console di gestione AWS e apri la console Lambda. 2. Nel pannello di navigazione, scegli Funzioni, quindi scegli la funzione che hai creato in precedenza. 3. Nella sezione Panoramica della funzione, scegli Aggiungi trigger. 4. Nella pagina Aggiungi trigger, nella sezione Configurazione Trigger, scegli CloudWatch Registri, quindi scegli Aggiungi. 5. Per Gruppo di log, scegli il gruppo di CloudWatch log che hai creato in precedenza. 6. Per Nome del filtro, inserisci un nome per il filtro. 7. Scegliere Aggiungi. 8. Nella scheda Configurazione della pagina della funzione, nella sezione Trigger, seleziona il trigger che hai appena aggiunto, quindi scegli Abilita. Per ulteriori informazioni, consulta Using Lambda	Amministratore cloud

Operazione	Description	Competenze richieste
	with CloudWatch Logs nella documentazione di Lambda.	

Operazione	Description	Competenze richieste
Aggiungi le autorizzazioni di pubblicazione SNS.	Aggiungi l'autorizzazione SNS:Publish al ruolo di esecuzione Lambda, in modo che Lambda possa effettuare chiamate API per pubblicare messaggi su SNS. 1. Trova il ruolo di esecuzione della funzione Lambda che hai creato in precedenza. 2. Aggiungi la seguente policy al tuo ruolo AWS Identity and Access Management (IAM): `{ "Version": "2012-10-17", "Statement": [ { "Sid": "AllowSNSPublish", "Effect": "Allow", "Action": [ "sns:GetTopicAttributes", "sns:Subscribe", "sns:Unsubscribe", "sns:Publish" ], "Resource": "*" }`	Amministratore del cloud

Operazione	Description	Competenze richieste
	] }	

Verifica la funzionalità della tua notifica SNS

Operazione	Description	Competenze richieste
Invia traffico tramite Network Firewall.	1. Invia o attendi che il traffico HTTPS passi attraverso Network Firewall. 2. Controlla l'e-mail di notifica SNS che ricevi da AWS quando il traffico attravers a Network Firewall. L'e-mail include i dettagli SNI per il traffico in uscita. Ad esempio, l'e-mail generata dal codice Lambda precedente avrà il seguente contenuto se il nome di dominio a cui si accede è https://aws.amazon.come il protocollo di abbonamento è EMAIL-JSON: { "Type": "Notification", "MessageId": "<messageID>", "TopicArn": "arn:aws:sns:us-west-2:123456789:testSNSTopic",	Tecnico collaudatore

Operazione	Description	Competenze richieste
	"Subject": "Server Name passed through the Network Firewall", "Message": "{\"ServerName\": \"Domain 'aws.amazon.com' accessed via AWS Network Firewall 'AWS-Network-Firewall-Multi-AZ-firewall\"}", "Timestamp": "2022-03-22T04:10:04.217Z", "SignatureVersion": "1", "Signature": "<Signature>", "SigningCertURL": "<SigningCertUrl>", "UnsubscribeURL": "<UnsubscribeURL>" }	
	Quindi, controlla il registro degli avvisi di Network Firewall in Amazon CloudWatch seguendo le istruzioni nella CloudWatch documentazione di Amazon. Il registro degli avvisi mostra il seguente output:	
	{ "firewall_name": "AWS-Network-Firewall-Multi-AZ-firewall", "availability_zone": "us-east-2b",	

Operazione	Description	Competenze richieste
	"event_timestamp": "<event timestamp>", "event": { "timestamp": "2021-03-22T04:10:04.214222+0000", "flow_id": <flow ID>, "event_type": "alert", "src_ip": "10.1.3.76", "src_port": 22761, "dest_ip": "99.86.59.73", "dest_port": 443, "proto": "TCP", "alert": { "action": "allowed", "signature_id": 2, "rev": 0, "signature": "", "category": "", "severity": 3 }, "tls": { "subject": "CN=aws.amazon.com", "issuerdn": "C=US, O=Amazon, OU=Server CA 1B, CN=Amazon", "serial": "<serial number>",	

Operazione	Description	Competenze richieste
	"fingerpr int": "<fingerprint ID>", "sni": "aws.amazon.com", "version": "TLS 1.2", "notbefor e": "2020-09-30T00:00: 00", "notafter ": "2021-09-23T12:00: 00", "ja3": {}, "ja3s": {} }, "app_proto": "tls" } }	

Usa Terraform per abilitare automaticamente Amazon GuardDuty per un'organizzazione

Aarthi Kannan, Amazon Web Services

Riepilogo

Amazon monitora GuardDuty continuamente i tuoi account Amazon Web Services (AWS) e utilizza l'intelligence sulle minacce per identificare attività impreviste e potenzialmente dannose all'interno del tuo ambiente AWS. L'abilitazione manuale GuardDuty di più account o organizzazioni, in più regioni AWS o tramite la Console di gestione AWS può essere complicata. Puoi automatizzare il processo utilizzando uno strumento Infrastructure as Code (IaC), come Terraform, che può fornire e gestire servizi e risorse multiaccount e multiregione nel cloud.

AWS consiglia di utilizzare AWS Organizations per configurare e gestire più account in GuardDuty. Questo modello è conforme a tale raccomandazione. Uno dei vantaggi di questo approccio è che, quando vengono creati o aggiunti nuovi account all'organizzazione, GuardDuty verranno abilitati automaticamente in questi account per tutte le regioni supportate, senza la necessità di un intervento manuale.

Questo modello dimostra come utilizzare HashiCorp Terraform per abilitare Amazon GuardDuty per tre o più account Amazon Web Services (AWS) in un'organizzazione. Il codice di esempio fornito con questo pattern esegue le seguenti operazioni:

- Abilita tutti GuardDuty gli account AWS che sono attualmente membri dell'organizzazione di destinazione in AWS Organizations
- Attiva la funzionalità Auto-Enable in GuardDuty, che abilita automaticamente tutti GuardDuty gli account che verranno aggiunti all'organizzazione di destinazione in futuro
- Consente di selezionare le regioni in cui si desidera abilitare GuardDuty
- Utilizza l'account di sicurezza dell'organizzazione come GuardDuty amministratore delegato
- Crea un bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) nell'account di registrazione e GuardDuty configura la pubblicazione dei risultati aggregati di tutti gli account in questo bucket
- Assegna una politica del ciclo di vita che trasferisce i risultati dal bucket S3 allo storage Amazon S3 Glacier Flexible Retrieval dopo 365 giorni, per impostazione predefinita

Puoi eseguire manualmente questo codice di esempio oppure integrarlo nella tua pipeline di integrazione continua e distribuzione continua (CI/CD).

Destinatari

Questo modello è consigliato agli utenti che hanno esperienza con Terraform, Python e AWS GuardDuty Organizations.

Prerequisiti e limitazioni

Prerequisiti

- Un account AWS attivo.
- Un'organizzazione è configurata in AWS Organizations e contiene almeno i seguenti tre account:
  - Un account di gestione: questo è l'account da cui si distribuisce il codice Terraform, autonomo o come parte della pipeline. CI/CD Lo stato Terraform è memorizzato anche in questo account.
  - Un account di sicurezza: questo account viene utilizzato come amministratore GuardDuty delegato. Per ulteriori informazioni, vedere Considerazioni importanti per gli amministratori GuardDuty delegati (documentazione). GuardDuty
  - Un account di registrazione: questo account contiene il bucket S3 in cui vengono GuardDuty pubblicati i risultati aggregati di tutti gli account membri.

  Per ulteriori informazioni su come configurare l'organizzazione con la configurazione richiesta, consulta Creare una struttura di account (AWS Well-Architected Labs).
- Un bucket Amazon S3 e una tabella Amazon DynamoDB che fungono da backend remoto per archiviare lo stato di Terraform nell'account di gestione. Per ulteriori informazioni sull'utilizzo dei backend remoti per lo stato Terraform, consulta S3 Backend (documentazione Terraform). Per un esempio di codice che configura la gestione remota dello stato con un backend S3, vedi 3-backend (Terraform Registry). remote-state-s Si notino i requisiti seguenti:
  - Il bucket S3 e la tabella DynamoDB devono trovarsi nella stessa regione.
  - Quando si crea la tabella DynamoDB, la chiave di partizione deve **`LockID`** essere (distinzione tra maiuscole e minuscole) e il tipo di chiave di partizione deve essere String. Tutte le altre impostazioni della tabella devono avere i valori predefiniti. Per ulteriori informazioni, vedere Informazioni sulle chiavi primarie e Creazione di una tabella (documentazione di DynamoDB).
- Un bucket S3 che verrà utilizzato per archiviare i log di accesso per il bucket S3 in cui verranno pubblicati i risultati. GuardDuty Per ulteriori informazioni, consulta Abilitazione della registrazione degli accessi al server Amazon S3 (documentazione Amazon S3). Se stai effettuando la

distribuzione in una landing zone di AWS Control Tower, puoi riutilizzare il bucket S3 nell'account di archiviazione dei log per questo scopo.

- La versione 0.14.6 o successiva di Terraform è installata e configurata. Per ulteriori informazioni, consulta Get Started — AWS (documentazione Terraform).
- Python versione 3.9.6 o successiva è installata e configurata. Per ulteriori informazioni, consulta Source releases (sito Web Python).
- AWS SDK per Python (Boto3) è installato. Per ulteriori informazioni, consulta Installazione (documentazione Boto3).
- jq è installato e configurato. Per ulteriori informazioni, consulta Download jq (documentazione jq).

Limitazioni

- Questo modello supporta i sistemi operativi macOS e Amazon Linux 2. Questo modello non è stato testato per l'uso nei sistemi operativi Windows.

> **Note**
>
> Amazon Linux 2 sta per terminare il supporto. Per ulteriori informazioni, consulta Amazon Linux 2 FAQs.

- GuardDuty non deve essere già abilitato in nessuno degli account, in nessuna delle regioni di destinazione.
- La soluzione IaC in questo modello non implementa i prerequisiti.
- Questo modello è progettato per una landing zone AWS che aderisce alle seguenti best practice:
  - La landing zone è stata creata utilizzando AWS Control Tower.
  - Account AWS separati vengono utilizzati per la sicurezza e la registrazione.

Versioni del prodotto

- Terraform versione 0.14.6 o successiva. Il codice di esempio è stato testato per la versione 1.2.8.
- Python versione 3.9.6 o successiva.

Architecture

Questa sezione offre una panoramica di alto livello di questa soluzione e dell'architettura stabilita dal codice di esempio. Il diagramma seguente mostra le risorse distribuite tra i vari account dell'organizzazione, all'interno di una singola regione AWS.

1. Terraform crea il ruolo GuardDutyTerraformOrgRoleAWS Identity and Access Management (IAM) nell'account di sicurezza e nell'account di registrazione.
2. Terraform crea un bucket S3 nella regione AWS predefinita nell'account di registrazione. Questo bucket viene utilizzato come destinazione di pubblicazione per aggregare tutti i GuardDuty risultati in tutte le regioni e provenienti da tutti gli account dell'organizzazione. Terraform crea anche una chiave AWS Key Management Service (AWS KMS) nell'account di sicurezza che viene utilizzata per crittografare i risultati nel bucket S3 e configura l'archiviazione automatica dei risultati dal bucket S3 nello storage S3 Glacier Flexible Retrieval.
3. Dall'account di gestione, Terraform designa l'account di sicurezza come amministratore delegato per. GuardDuty Ciò significa che l'account di sicurezza ora gestisce il GuardDuty servizio per tutti gli account dei membri, incluso l'account di gestione. Gli account dei singoli membri non possono essere sospesi o GuardDuty disattivati da soli.
4. Terraform crea il GuardDuty rilevatore nell'account di sicurezza, per l' GuardDuty amministratore delegato.
5. Se non è già abilitato, Terraform abilita la protezione S3. GuardDuty Per ulteriori informazioni, consulta la protezione di Amazon S3 in Amazon GuardDuty (GuardDuty documentazione).
6. Terraform registra tutti gli account membri attuali e attivi dell'organizzazione come membri. GuardDuty
7. Terraform configura l'amministratore GuardDuty delegato per pubblicare i risultati aggregati di tutti gli account membri nel bucket S3 nell'account di registrazione.
8. Terraform ripete i passaggi da 3 a 7 per ogni regione AWS scelta.

Automazione e scalabilità

Il codice di esempio fornito è modularizzato in modo da poterlo integrare nella CI/CD pipeline per l'implementazione automatizzata.

Tools (Strumenti)

Servizi AWS

- Amazon DynamoDB è un servizio di database NoSQL interamente gestito che offre prestazioni elevate, prevedibili e scalabili.
- Amazon GuardDuty è un servizio di monitoraggio continuo della sicurezza che analizza ed elabora i log per identificare attività impreviste e potenzialmente non autorizzate nel tuo ambiente AWS.
- AWS Identity and Access Management (IAM) ti aiuta a gestire in modo sicuro l'accesso alle tue risorse AWS controllando chi è autenticato e autorizzato a utilizzarle.
- AWS Key Management Service (AWS KMS) ti aiuta a creare e controllare chiavi crittografiche per proteggere i tuoi dati.
- AWS Organizations è un servizio di gestione degli account che ti aiuta a consolidare più account AWS in un'organizzazione da creare e gestire centralmente.
- Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione degli oggetti basato sul cloud che consente di archiviare, proteggere e recuperare qualsiasi quantità di dati.
- AWS SDK for Python (Boto3) è un kit di sviluppo software che ti aiuta a integrare l'applicazione, la libreria o lo script Python con i servizi AWS.

Altri strumenti e servizi

- HashiCorp Terraform è un'applicazione di interfaccia a riga di comando che consente di utilizzare il codice per fornire e gestire l'infrastruttura e le risorse cloud.
- Python è un linguaggio di programmazione generico.
- jq è un processore a riga di comando che consente di lavorare con i file JSON.

Deposito di codice

Il codice per questo pattern è disponibile GitHub nel organizations-with-terraform repository amazon-guardduty-for-aws-.

Epiche

Abilita GuardDuty nell'organizzazione

Operazione	Description	Competenze richieste
Clonare il repository.	In una shell Bash, esegui il seguente comando. In Clona il repository nella sezione Informazioni aggiuntive, puoi copiare il comando completo contenente l'URL del repository. GitHub Questo clona il repository - da. amazon-guardduty-for-aws organizations-with-terraform GitHub `git clone <github-repository-url>`	DevOps ingegnere
Modifica il file di configurazione Terraform.	1. Nella `root` cartella del repository clonato, replica il file configuration.json .sample eseguendo il seguente comando. `cp configuration.json .sample configuration.json` 2. Modifica il nuovo file configuration.json e definisci i valori per ciascuna delle seguenti variabili: • `management_acc_id` — ID dell'account di gestione.	DevOps ingegnere, AWS generale, Terraform, Python

Operazione	Description	Competenze richieste
	• `delegated_admin_acc_id` — ID dell'account di sicurezza. • `logging_acc_id` — ID dell'account di registrazione. • `target_regions` — Elenco separato da virgole delle regioni AWS in cui desideri abilitare. GuardDuty • `organization_id` — ID AWS Organizations dell'organizzazione in cui stai abilitando GuardDuty. • `default_region` — La regione in cui è archiviato lo stato di Terraform nell'account di gestione. Questa è la stessa regione in cui hai distribuito il bucket S3 e la tabella DynamoDB per il backend Terraform. • `role_to_assume_for_role_creation` — Nome da assegnare a un nuovo ruolo IAM negli account di sicurezza e registrazione. Creerai questo nuovo ruolo nella prossima storia. Terraform assume questo	

Operazione	Description	Competenze richieste
	ruolo per creare il ruolo `GuardDutyTerraformOrgRole` IAM negli account di sicurezza e registrazione. • `finding_publishing_frequency` — Frequenza con cui vengono GuardDuty pubblicati i risultati nel bucket S3. • `guardduty_findings_bucket_region` — Regione preferita in cui si desidera creare il bucket S3 per i risultati pubblicati. • `logging_acc_s3_bucket_name` — Nome preferito per il bucket S3 per i risultati pubblicati. • `security_acc_kms_key_alias` — Alias AWS KMS per la chiave utilizzata per crittografare i risultati. GuardDuty • `s3_access_log_bucket_name` — Nome di un bucket S3 preesistente in cui desideri raccogliere i log di accesso per il bucket S3 utilizzato per i risultati. GuardDuty Questo bucket dovrebbe	

Operazione	Description	Competenze richieste
	trovarsi nella stessa regione AWS del bucket dei GuardDuty risultati. • `tfm_state_backend_s3_bucket` — Nome del bucket S3 preesistente per memorizzare lo stato del backend remoto Terraform. • `tfm_state_backend_dynamodb_table` — Nome della tabella DynamoDB preesistente per il blocco dello stato Terraform. 3. Salva e chiudi il file di configurazione .	

Operazione	Description	Competenze richieste
Genera CloudFormation modelli per nuovi ruoli IAM.	Questo modello include una soluzione IaC per creare due CloudFormation modelli. Questi modelli creano due ruoli IAM che Terraform utilizza durante il processo di configurazione. Questi modelli aderiscono alle migliori pratiche di sicurezza delle autorizzazioni con privilegi minimi. 1. In una shell Bash, nella cartella del repository, vai a. `root cfn-templates/` Questa cartella contiene file CloudFormation di modelli con stub. 2. Eseguire il seguente comando seguente. Questo sostituisce gli stub con i valori forniti nel file configuration.json. `bash scripts/replace_config_stubs.sh` 3. Verifica che i seguenti CloudFormation modelli siano stati creati nella cartella: `cfn-templates/` • management-account-role.yaml: questo file	DevOps ingegnere, General AWS

Operazione	Description	Competenze richieste
	contiene la definizione del ruolo e le autorizza zioni associate per il ruolo IAM nell'acco unt di gestione, che dispone delle autorizza zioni minime richieste per completare questo modello. • role-to-assume-for-role-creation.yaml — Questo file contiene la definizio ne del ruolo e le autorizza zioni associate per il ruolo IAM negli account di sicurezza e registraz ione. Terraform assume questo ruolo per creare il ruolo in questi account. GuardDutyTerraform OrgRole	

Operazione	Description	Competenze richieste
Crea i ruoli IAM.	Seguendo le istruzioni in Creazione di uno stack (CloudFormation documentazione), procedi come segue: 1. Distribuisci lo stack role-to-assume-for-role-creation.yaml sia nell'account di sicurezza che in quello di registrazione. 2. Distribuisci lo stack management-account-role.yaml nell'account di gestione. Dopo aver creato correttamente lo stack e visto lo stato dello `CREATE_COMPLETE` stack, nell'output, prendi nota dell'Amazon Resource Name (ARN) di questo nuovo ruolo.	DevOps ingegnere, General AWS
Assumi il ruolo IAM nell'account di gestione.	Come best practice in materia di sicurezza, ti consigliamo di assumere il nuovo ruolo management-account-roleIAM prima di procedere. In AWS Command Line Interface (AWS CLI), inserisci il comando in Assumi il ruolo IAM dell'account di gestione nella sezione Informazioni aggiuntive.	DevOps ingegnere, General AWS

Operazione	Description	Competenze richieste
Esegui lo script di configura zione.	Nella `root` cartella del repository, esegui il comando seguente per avviare lo script di installazione. `bash scripts/full-setup .sh` Lo script full-setup.sh esegue le seguenti azioni: • Esporta tutti i valori di configurazione come variabili di ambiente • Genera i file di codice backend.tf e terraform.tfvars per ogni modulo Terraform • Consente l'accesso affidabil e GuardDuty all'interno dell'organizzazione tramite la CLI di AWS. • Importa lo stato dell'orga nizzazione nello stato Terraform • Crea il bucket S3 per la pubblicazione dei risultati nell'account di registrazione • Crea la chiave AWS KMS per crittografare i risultati nell'account di sicurezza • Abilita GuardDuty in tutta l'organizzazione, in tutte le regioni selezionate, come	DevOps ingegnere, Python

Operazione	Description	Competenze richieste
	descritto nella sezione Architettura	

(Facoltativo) Disabilita GuardDuty nell'organizzazione

Operazione	Description	Competenze richieste
Esegui lo script di pulizia.	Se hai utilizzato questo schema GuardDuty per abilitare l'organizzazione e desideri disabilitarlo GuardDuty, nella `root` cartella del repository, esegui il comando seguente per avviare lo script cleanup-gd.sh. `bash scripts/cleanup-gd.sh` Questo script si disabilita GuardDuty nell'organizzazione di destinazione, rimuove tutte le risorse distribuite e ripristin a l'organizzazione allo stato precedente prima di utilizzar e Terraform per l'attivazione. GuardDuty **Note** Questo script non rimuove i file di stato Terraform né blocca i file dai backend locali	DevOps ingegnere, AWS generale, Terraform, Python

Operazione	Description	Competenze richieste
	e remoti. Se necessari o, è necessario eseguire queste azioni manualmen te. Inoltre, questo script non elimina l'organizzazione importata o gli account da essa gestiti. L'accesso affidabile per GuardDuty non è disabilitato come parte dello script di pulizia.	
Rimuovi i ruoli IAM.	Elimina gli stack creati con i modelli role-to-assume-for -role-creation.yaml e .yaml. management-account- role CloudFormation Per ulteriori informazioni, consulta Eliminazione di uno stack (documentazione). CloudForm ation	DevOps ingegnere, General AWS

Risorse correlate

Documentazione AWS

- Gestione di più account (GuardDuty documentazione)
- Concessione del privilegio minimo (documentazione IAM)

Marketing AWS

- Amazon GuardDuty

- AWS Organizations

Altre risorse

- Terraform
- Documentazione CLI Terraform

Informazioni aggiuntive

Clona il repository

Esegui il comando seguente per clonare il repository. GitHub

```
git clone https://github.com/aws-samples/amazon-guardduty-for-aws-organizations-with-terraform
```

Assumi il ruolo IAM dell'account di gestione

Per assumere il ruolo IAM nell'account di gestione, esegui il comando seguente. Sostituisci <IAM role ARN> con l'ARN del ruolo IAM.

```
export ROLE_CREDENTIALS=$(aws sts assume-role --role-arn <IAM role ARN> --role-session-name AWSCLI-Session --output json)
export AWS_ACCESS_KEY_ID=$(echo $ROLE_CREDENTIALS | jq .Credentials.AccessKeyId | sed 's/"//g')
export AWS_SECRET_ACCESS_KEY=$(echo $ROLE_CREDENTIALS | jq .Credentials.SecretAccessKey | sed 's/"//g')
export AWS_SESSION_TOKEN=$(echo $ROLE_CREDENTIALS | jq .Credentials.SessionToken | sed 's/"//g')
```

Verifica le best practice operative per PCI DSS 4.0 utilizzando AWS Config

Tala Qraitem e Alex Goff, Amazon Web Services

Riepilogo

Il Payment Card Industry Data Security Standard (PCI DSS) delinea i protocolli tecnici e operativi essenziali per contribuire a salvaguardare i dati di pagamento. Il PCI DSS è stato sviluppato per incoraggiare e migliorare la sicurezza dei dati per i conti delle carte di pagamento. Inoltre, facilita l'adozione globale di misure di sicurezza coerenti. Sebbene sia progettato specificamente per ambienti con dati relativi ai conti delle carte di pagamento, puoi utilizzare PCI DSS per proteggerti dalle minacce e proteggere altri elementi dell'ecosistema dei pagamenti.

La versione PCI DSS 4.0 è stata rilasciata per soddisfare i requisiti in evoluzione, fornire chiarimenti o indicazioni aggiuntive e migliorare la struttura e il formato dello standard. Per ulteriori informazioni sulle modifiche, vedere Riepilogo delle modifiche rispetto alla versione PCI DSS da 3.2.1 a 4.0.

Un pacchetto di AWS Config conformità è una raccolta di AWS Config regole e azioni correttive che aiutano a creare controlli di governance relativi alla sicurezza, all'operatività o all'ottimizzazione dei costi. È possibile distribuire un pacchetto di conformità come singola entità in un Account AWS and Regione AWS oppure è possibile distribuirlo all'interno di un'organizzazione in. AWS Organizations

I pacchetti di conformità per PCI DSS versione 4.0 ampliano e si basano sul pacchetto di conformità per la versione 3.2.1. Le regole del pacchetto di conformità corrispondono alle regole dello standard. Per ulteriori informazioni, consulta la mappatura fornita nella sezione Allegati. È possibile scegliere tra due versioni di questo pacchetto di conformità: una che include i tipi di risorse globali e una che li esclude.

> **Important**
>
> I pacchetti di conformità non sono progettati per garantire la piena conformità a uno specifico standard di governance o conformità. È responsabilità dell'utente valutare autonomamente se l'utilizzo soddisfa i requisiti legali e normativi applicabili.

Prerequisiti e limitazioni

Prerequisiti

- Avere un attivo Account AWS.
- Configurare AWS Config.
- Soddisfa i prerequisiti per i pacchetti di conformità.
- Implementa il pacchetto di conformità PCI DSS versione 3.2.1.
- Disponi delle autorizzazioni per accedere e gestire i pacchetti di conformità. AWS Config Per un esempio di politica, consulta la sezione Informazioni aggiuntive di questo modello.

Limitazioni

- Your Account AWS ha delle quote predefinite, precedentemente denominate limiti, per ciascuna di esse. Servizio AWS Salvo diversa indicazione, ogni quota si applica a una Regione specifica. È possibile richiedere aumenti per alcune quote, ma non tutte le quote possono essere aumentate. Assicurati di conoscere i limiti del AWS Config servizio, inclusi i limiti per i pacchetti di conformità per account singolo e i pacchetti di conformità per le organizzazioni.
- La versione di questo pacchetto di conformità che include i tipi di risorse globali è destinata alla distribuzione solo nella regione. `us-east-1`
- La versione di questo pacchetto di conformità che esclude i tipi di risorse globali è destinata alla distribuzione solo nelle seguenti regioni:
  - `ap-east-1`
  - `ap-south-1`
  - `ap-northeast-2`
  - `ap-southeast-1`
  - `ap-southeast-2`
  - `ap-northeast-1`
  - `ca-central-1`
  - `eu-central-1`
  - `eu-west-1`
  - `eu-west-2`
  - `eu-west-3`
  - `eu-north-1`
  - `sa-east-1`
  - `us-east-2`

- `us-west-1`
- `us-west-2`

Tools (Strumenti)

Servizi AWS

- AWS Configfornisce una visualizzazione dettagliata delle risorse disponibili Account AWS e di come sono configurate. Ti aiuta a identificare in che modo le risorse sono correlate tra loro e in che modo le loro configurazioni sono cambiate nel tempo.
- AWS Systems Managerconsente di gestire le applicazioni e l'infrastruttura in esecuzione in. Cloud AWS Semplifica la gestione delle applicazioni e delle risorse, riduce i tempi di rilevamento e risoluzione dei problemi operativi e aiuta a gestire le AWS risorse in modo sicuro su larga scala.

Archivio di codici

I conformance pack si trovano nell'archivio dei AWS Config conformance pack. GitHub Questo repository contiene i seguenti modelli relativi alla versione PCI DSS 4.0:

- Migliori pratiche operative per PCI-DSS-v4.0- .yaml including-global-resourcetypes
- Migliori pratiche operative per-PCI-DSS-v4.0- .yaml excluding-global-resourcetypes

Epiche

Implementa e gestisci il pacchetto di conformità

Operazione	Description	Competenze richieste
Scarica il pacchetto di conformità.	Se stai distribuendo il pacchetto di conformità nella `us-east-1` regione, scarica il modello Operational-Best-Practices-for-PCI-DSS-v4.0- .yaml. including-global-resourcetypes	DevOps ingegnere

Operazione	Description	Competenze richieste
	Se stai distribuendo il conformance pack in un'altra regione, scarica il excluding -global-resourcetypesmodello Operational-Best-Practices-for-PCI-DSS-v4.0- .yaml.	
(Facoltativo) Modificare il pacchetto di conformità.	È possibile modificare il modello del pacchetto di conformità in base alle esigenze specifiche dell'organizzazione. Ad esempio, è possibile creare azioni correttive personalizzate. Per ulteriori informazioni su come creare e modificare i modelli, vedere Creazione di modelli per pacchetti di conformità personalizzati nella documentazione. AWS Config	Informazioni generali su AWS

Operazione	Description	Competenze richieste
Implementa il pacchetto di conformità.	Se stai implementando in un target Account AWS o Regione AWS segui le istruzioni in Distribuzione dei conformance pack nella documentazione. AWS Config Puoi usare il Console di gestione AWS o il (). AWS Command Line Interface AWS CLI Se stai distribuendo il pacchetto di conformità in un'organizzazione in AWS Organizations, segui le istruzioni in Distribuisci il pacchetto di AWS Config conformità utilizzando Quick Setup nella documentazione. AWS Systems Manager	Informazioni generali su AWS
(Facoltativo) Modifica il pacchetto di conformità.	Se desideri modificare il pacchetto di conformità, segui le istruzioni in Modifica dei pacchetti di conformità nella documentazione. AWS Config È possibile utilizzare o il. Console di gestione AWS AWS CLI	Informazioni generali su AWS

Operazione	Description	Competenze richieste
(Facoltativo) Eliminare il pacchetto di conformità.	Se desideri eliminare il pacchetto di conformità, segui le istruzioni in Eliminazione dei pacchetti di conformità nella documentazione. AWS Config È possibile utilizzare il o il. Console di gestione AWS AWS CLI	Informazioni generali su AWS

Risorse correlate

AWS resources

- Pacchetti di conformità per AWS Config(documentazione)AWS Config
- Implementazione del AWS Config Conformance Pack tramite Quick Setup (documentazione di Systems Manager)
- Conformità allo standard PCI DSS su (sito Web) AWSAWS
- PCI DSS versione 4.0 attiva (guida alla AWS conformità)

Risorse PCI DSS

- PCI DSS versione 4.0 Resource Hub
- Libreria di documenti del PCI Security Standards Council
- Riepilogo delle modifiche rispetto alla versione PCI DSS da 3.2.1 a 4.0

Informazioni aggiuntive

Di seguito è riportato un esempio di policy AWS Identity and Access Management (IAM) che consente all'utente di accedere AWS Config e gestire i pacchetti di conformità:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
```

```
            "Effect": "Allow",
            "Action": [
                "config:PutConfigRule",
                "config:PutConformancePack",
                "config:DeleteConfigRule",
                "config:DeleteRemediationConfiguration",
                "config:DeleteConformancePack",
                "config:PutRemediationConfigurations",
                "config:BatchGetAggregateResourceConfig",
                "config:BatchGetResourceConfig",
                "config:Get*",
                "config:Describe*",
                "config:Deliver*",
                "config:List*",
                "config:Select*"
            ],
            "Resource": "*"
        }
    ]
}
```

Allegati

Per accedere al contenuto aggiuntivo associato a questo documento, decomprimi il seguente file: attachment.zip

Altri modelli

- Accedi a un host bastion utilizzando Session Manager e Amazon EC2 Instance Connect
- Accedi alle applicazioni container in modo privato su Amazon ECS utilizzando AWS Fargate, PrivateLink AWS e un Network Load Balancer
- Accedi alle applicazioni container in modo privato su Amazon ECS utilizzando AWS PrivateLink e un Network Load Balancer
- Accedi alle applicazioni container in modo privato su Amazon EKS utilizzando AWS PrivateLink e un Network Load Balancer
- Consenti alle EC2 istanze l'accesso in scrittura ai bucket S3 negli account AMS
- Associa un AWS CodeCommit repository in uno Account AWS ad Amazon SageMaker AI Studio Classic in un altro account
- Autentica gli utenti delle applicazioni React esistenti utilizzando Amazon Cognito e l'interfaccia utente AWS Amplify
- Automatizza l'aggiunta o l'aggiornamento delle voci di registro di Windows utilizzando AWS Systems Manager
- Automatizza l'applicazione della crittografia in AWS Glue utilizzando un modello AWS CloudFormation
- Associa automaticamente una policy gestita da AWS per Systems Manager ai profili di EC2 istanza utilizzando Cloud Custodian e AWS CDK
- Crittografa automaticamente i volumi Amazon EBS esistenti e nuovi
- Blocca l'accesso pubblico ad Amazon RDS utilizzando Cloud Custodian
- Centralizza la risoluzione DNS utilizzando AWS Managed Microsoft AD e localmente Microsoft Active Directory
- Implementa la scansione Checkov centralizzata e personalizzata per applicare le policy prima di implementare l'infrastruttura AWS
- Verifica la presenza di tag obbligatori EC2 nelle istanze al momento del lancio
- Configurazione dell'accesso multi-account in Amazon DynamoDB
- Configurare la crittografia HTTPS per Oracle JD Edwards EnterpriseOne su Oracle WebLogic utilizzando un Application Load Balancer
- Configura l'autenticazione TLS reciproca per le applicazioni in esecuzione su Amazon EKS
- Configurare l'autenticazione Windows per Amazon RDS for Microsoft SQL Server utilizzando AWS Managed Microsoft AD

- Connect utilizzando un tunnel SSH in pGAdmin
- Crea un'app React utilizzando AWS Amplify e aggiungi l'autenticazione con Amazon Cognito
- Crea un rapporto sui risultati di Network Access Analyzer per l'accesso a Internet in entrata in più Account AWS
- Personalizza CloudWatch gli avvisi Amazon per AWS Network Firewall
- Implementa una ChatOps soluzione per gestire i risultati della scansione SAST utilizzando Amazon Q Developer nelle applicazioni di chat, azioni personalizzate e CloudFormation
- Implementa la convalida della sicurezza della codifica in tempo reale utilizzando un server MCP con Kiro e altri assistenti di codifica
- Documenta il progetto della tua landing zone AWS
- Abilita connessioni crittografate per le istanze DB PostgreSQL in Amazon RDS
- Crittografa un'istanza database Amazon RDS for PostgreSQL esistente
- Applica il tagging automatico dei database Amazon RDS al momento del lancio
- Applica l'etichettatura dei cluster Amazon EMR al momento del lancio
- Assicurati che la registrazione di Amazon EMR su Amazon S3 sia abilitata al momento del lancio
- Genera un CloudFormation modello AWS contenente le regole gestite di AWS Config utilizzando Troposphere
- Ricevi notifiche Amazon SNS quando lo stato chiave di una chiave AWS KMS cambia
- Aiutaci a far rispettare il tagging di DynamoDB
- Identifica e avvisa quando le risorse Amazon Data Firehose non sono crittografate con una chiave AWS KMS
- Implementa l'autenticazione basata su Microsoft Entra ID in un'applicazione mainframe modernizzata AWS Blu Age
- Implementa l'autenticazione SAML 2.0 per Amazon WorkSpaces utilizzando Auth0 e AWS Managed Microsoft AD
- SHA1 Implementa l'hashing per i dati PII durante la migrazione da SQL Server a PostgreSQL
- Migliora le prestazioni operative abilitando Amazon DevOps Guru su più regioni AWS, account e OUs con AWS CDK
- Acquisisci e migra istanze EC2 Windows in un account AWS Managed Services
- Esegui la migrazione da Amazon RDS for Oracle ad Amazon RDS for PostgreSQL in modalità SSL utilizzando AWS DMS
- Migrazione di uno stack ELK su Elastic Cloud su AWS

- Esegui la migrazione di un carico di lavoro F5 BIG-IP su F5 BIG-IP VE su Cloud AWS
- Monitora Amazon Aurora per le istanze senza crittografia
- Fornisci ruoli IAM con privilegi minimi implementando una soluzione di distribuzione automatica di ruoli
- Proteggi e semplifica l'accesso degli utenti in un database federativo Db2 su AWS utilizzando contesti affidabili
- Invia AWS WAF i log a Splunk utilizzando AWS Firewall Manager Amazon Data Firehose
- Distribuisci contenuti statici in un bucket Amazon S3 tramite un VPC utilizzando Amazon CloudFront
- Configura end-to-end la crittografia per le applicazioni su Amazon EKS utilizzando cert-manager e Let's Encrypt
- Usa l'utente IDs nelle policy IAM per il controllo e l'automazione degli accessi
- Verificare che i sistemi di bilanciamento del carico ELB richiedano la terminazione TLS
- Visualizza i log e i parametri di AWS Network Firewall utilizzando Splunk
- Visualizza i report sulle credenziali IAM per tutti gli account AWS utilizzando Amazon Quick Sight

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.